Contemplating Statistics: estimation and regression according to arc lengths

by

Mattheüs Theodor Loots

Submitted in partial fulfilment of the requirements for the degree Philosophiae Doctor (Mathematical Statistics) in the Faculty of Natural and Agricultural Sciences University of Pretoria, Pretoria

September 2017

© University of Pretoria
Publication data:


Electronic, hyperlinked versions of this thesis are available online, as Adobe PDF files, at:

http://upetd.up.ac.za/UPeTD.htm
Contemplating Statistics: estimation and regression according to arc lengths

by

Mattheüs Theodor Loots
E-mail: theodor.loots@up.ac.za

Abstract

Advances in computing has undoubtfully been one of the main catalysts in the formation of the discipline always known as Statistics. A fundamental question addressed here is whether computing facilities, such as parallel or high performance computing, could assist in the development of methodologies that render stronger results, based on some predetermined optimality criterion. The candidate at the hand of which this enquiry is made, is the arc length of some statistical function. Estimation, goodness-of-fit, linear regression and non-linear regression, which may all be considered as central themes in Statistics, are revisited, and redefined in terms of this new measure. The results resulting from these arc length methodologies are obtained from simulation, as well as from real case studies, and contrasted to that obtained using their classical counterparts. Mathematical premises for the proposed methods are provided, together with the documentation accompanying the companion R package, along with the data utilised for the applications.

Keywords: Arc length, goodness-of-fit, non-linear regression, parameter estimation, regression.

Supervisor : Prof. A. Bekker
Department : Department of Statistics
Degree : Philosophiae Doctor
“Use?” replied Reepicheep. “Use, Captain? If by use you mean filling our bellies or our purses, I confess it will be no use at all. So far as I know we did not set sail to look for things useful but to seek honour and adventures. And here is as great an adventure as ever I heard of, and here, if we turn back, no little impeachment of all our honours.”


“He who loves practise without theory is like the sailor who boards ship without a rudder and compass and never knows where he may cast.”

Leonardo da Vinci

To Katryn, Katelan, Matthew and Galen, with love.
Acknowledgements

Without the various contributions described in the following list, this work would not have been able to stand as it is today:

- To God, my creator: Thank You for providing the breath to sing Your song; You alone shall have my heart;
- My home, Katryn, Katelan, Matthew and Galen: Thank you for all the motivation and inspiration, and for keeping my feet on the earth;
- My natural and spiritual family: You kept me going;
- Mrs. Hestelle Viljoen: There once was a Mathematics teacher who taught me that potential only describes the void of things to come. Thank you for believing;
- Prof. Andriëtte Bekker: Thank you for sharing a journey very few others would have embarked upon;
- Colleagues at the Department of Statistics: I cannot list you all, but thank you for fourteen years of dedication and support;
- My co-authors, Dr. Steven Hussey and Prof. Walter Focke: Your problems made this thesis come to life;
- Prof. Brenda Wingfield: The mug;
- Examinors: Your attention to detail completed this work;
- The Centre for High Performance Computing: Thank you for providing access to Lengau, and especially to Mr. Dane Kennedy for your enthusiasm and support;
- This work is based on the research supported in part by the “National Research Foundation” of South Africa for the grant, Unique Grant No. 94108. Any opinion, finding and conclusion or recommendation expressed in this material is that of the author(s) and the NRF does not accept any liability in this regard.
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of Figures</td>
<td>v</td>
</tr>
<tr>
<td>List of Tables</td>
<td>vii</td>
</tr>
<tr>
<td>1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Motivation</td>
<td>3</td>
</tr>
<tr>
<td>1.2 Objectives</td>
<td>4</td>
</tr>
<tr>
<td>1.3 Contributions</td>
<td>4</td>
</tr>
<tr>
<td>1.4 Thesis Outline</td>
<td>5</td>
</tr>
<tr>
<td>2 Preliminaries</td>
<td>7</td>
</tr>
<tr>
<td>2.1 Mathematical Background</td>
<td>7</td>
</tr>
<tr>
<td>2.2 Implementation Infrastructure</td>
<td>8</td>
</tr>
<tr>
<td>2.3 Summary</td>
<td>9</td>
</tr>
<tr>
<td>3 Estimation</td>
<td>10</td>
</tr>
<tr>
<td>3.1 Implementation Infrastructure</td>
<td>11</td>
</tr>
<tr>
<td>3.2 Sample Arc Length Statistics</td>
<td>11</td>
</tr>
<tr>
<td>3.3 Parameter Estimation</td>
<td>15</td>
</tr>
<tr>
<td>3.4 Goodness-Of-Fit</td>
<td>18</td>
</tr>
<tr>
<td>3.4.1 Skewness Considerations</td>
<td>19</td>
</tr>
<tr>
<td>3.4.2 Mesokurtic Considerations</td>
<td>22</td>
</tr>
<tr>
<td>3.5 Summary</td>
<td>24</td>
</tr>
<tr>
<td>4 Arc Length Regression</td>
<td>26</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>27</td>
</tr>
</tbody>
</table>
4.2 Equating Moments ........................................... 28
4.3 Arc Length Regression ................................. 31
4.4 Implementation Infrastructure ....................... 31
4.5 Application ................................................. 32
  4.5.1 Moment Matching ................................. 33
  4.5.2 Arc Length Matching ........................... 35
4.6 Model Evaluation ......................................... 37
4.7 Summary .................................................. 39

5 Non-Linear Arc Length Regression .................. 41
  5.1 Introduction ............................................. 41
  5.2 Problem Description ................................. 42
  5.3 The Four-Parameter Kappa Distribution ........... 47
  5.4 Results .................................................. 52
    5.4.1 Implementation Infrastructure ................. 54
    5.4.2 Sample1 ........................................... 54
    5.4.3 Sample2 ........................................... 56
  5.5 Summary ................................................. 59

6 A Theoretical Framework for Arc Length Based Statistics 60
  6.1 Introduction ............................................. 60
  6.2 Uniform Distribution ................................. 64
  6.3 Exponential Distribution ............................ 64
  6.4 Summary ................................................. 66

7 Conclusions ................................................. 67
  7.1 Summary of Conclusions ............................. 67
  7.2 Future Work ............................................ 68

Bibliography .................................................. 70

A The alR Package .......................................... 78
  A.1 Estimation .............................................. 78
    A.1.1 Method of Arc Lengths ......................... 78
    alE ..................................................... 78
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.1.2 Arc Length Test</td>
<td>81</td>
</tr>
<tr>
<td>alEdist</td>
<td>81</td>
</tr>
<tr>
<td>A.2 Arc Length Regression</td>
<td>84</td>
</tr>
<tr>
<td>A.2.1 Moment Matching</td>
<td>84</td>
</tr>
<tr>
<td>mmKDEboot</td>
<td>84</td>
</tr>
<tr>
<td>A.2.2 Arc Length Matching</td>
<td>87</td>
</tr>
<tr>
<td>alKDEboot</td>
<td>87</td>
</tr>
<tr>
<td>A.2.3 Bhattacharyya Divergence Test</td>
<td>91</td>
</tr>
<tr>
<td>bhatt.test</td>
<td>91</td>
</tr>
<tr>
<td>A.3 Non-Linear Arc Length Regression</td>
<td>92</td>
</tr>
<tr>
<td>A.3.1 Non-linear Least Squares</td>
<td>92</td>
</tr>
<tr>
<td>kappa4nlsBoot</td>
<td>92</td>
</tr>
<tr>
<td>A.3.2 Non-Linear Arc Length Regression</td>
<td>95</td>
</tr>
<tr>
<td>kappa4alBoot</td>
<td>96</td>
</tr>
<tr>
<td>A.4 Summary</td>
<td>99</td>
</tr>
<tr>
<td>B Simulation Results for the Method of Arc Lengths</td>
<td>100</td>
</tr>
<tr>
<td>B.1 Introduction</td>
<td>100</td>
</tr>
<tr>
<td>B.2 Simulation Results for $N(-2,0.5)$</td>
<td>101</td>
</tr>
<tr>
<td>B.3 Simulation Results for $N(-2,1)$</td>
<td>112</td>
</tr>
<tr>
<td>B.4 Simulation Results for $N(-2,3.5)$</td>
<td>112</td>
</tr>
<tr>
<td>B.5 Simulation Results for $N(0,0.5)$</td>
<td>112</td>
</tr>
<tr>
<td>B.6 Simulation Results for $N(0,1)$</td>
<td>112</td>
</tr>
<tr>
<td>B.7 Simulation Results for $N(0,3.5)$</td>
<td>112</td>
</tr>
<tr>
<td>B.8 Simulation Results for $N(2,0.5)$</td>
<td>112</td>
</tr>
<tr>
<td>B.9 Simulation Results for $N(2,1)$</td>
<td>112</td>
</tr>
<tr>
<td>B.10 Simulation Results for $N(2,3.5)$</td>
<td>112</td>
</tr>
<tr>
<td>B.11 Summary</td>
<td>112</td>
</tr>
<tr>
<td>C Machine Learning</td>
<td>121</td>
</tr>
<tr>
<td>C.1 Random Forest</td>
<td>121</td>
</tr>
<tr>
<td>C.2 Support Vector Regression</td>
<td>122</td>
</tr>
<tr>
<td>C.3 Summary</td>
<td>123</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Violin plots for the empirical distributions of discrete arc length statistics</td>
<td>13</td>
</tr>
<tr>
<td>3.2</td>
<td>Violin plots for the empirical distributions of continuous arc length statistics</td>
<td>14</td>
</tr>
<tr>
<td>3.3</td>
<td>Estimated rMSE for $\mu$ and $\sigma$ for varying sample sizes</td>
<td>17</td>
</tr>
<tr>
<td>3.4</td>
<td>Heatmaps of $p$-values for GoF using arc lengths (skewness considerations)</td>
<td>20</td>
</tr>
<tr>
<td>3.5</td>
<td>GoF comparisons for skew alternatives</td>
<td>21</td>
</tr>
<tr>
<td>3.6</td>
<td>Heatmaps of $p$-values for GoF using arc lengths (mesokurtic deviations)</td>
<td>23</td>
</tr>
<tr>
<td>3.7</td>
<td>GoF comparisons for various mesokurtic deviances</td>
<td>24</td>
</tr>
<tr>
<td>4.1</td>
<td>Plots for transformed expression data resulting from OLS</td>
<td>33</td>
</tr>
<tr>
<td>4.2</td>
<td>Plots for transformed expression data resulting from method of moments</td>
<td>35</td>
</tr>
<tr>
<td>4.3</td>
<td>Plots for transformed expression data resulting from arc length regression</td>
<td>37</td>
</tr>
<tr>
<td>5.1</td>
<td>Illustration of the two methods for the evaluation of the IP</td>
<td>46</td>
</tr>
<tr>
<td>5.2</td>
<td>$(h, k)$ combinations for the four-parameter kappa distribution</td>
<td>51</td>
</tr>
<tr>
<td>5.3</td>
<td>Time vs. conductivity curves</td>
<td>53</td>
</tr>
<tr>
<td>5.4</td>
<td>Fitted sigmoidal curves to sample1</td>
<td>56</td>
</tr>
<tr>
<td>5.5</td>
<td>Fitted sigmoidal curves to sample2</td>
<td>58</td>
</tr>
<tr>
<td>B.1</td>
<td>Estimated rMSE for $\mu = -2$ and $\sigma = 0.5$ for varying sample sizes</td>
<td>102</td>
</tr>
<tr>
<td>B.2</td>
<td>Estimated rMSE for $\mu = -2$ and $\sigma = 1$ for varying sample sizes</td>
<td>103</td>
</tr>
<tr>
<td>B.3</td>
<td>Estimated rMSE for $\mu = -2$ and $\sigma = 3.5$ for varying sample sizes</td>
<td>104</td>
</tr>
<tr>
<td>B.4</td>
<td>Estimated rMSE for $\mu = 0$ and $\sigma = 0.5$ for varying sample sizes</td>
<td>105</td>
</tr>
<tr>
<td>B.5</td>
<td>Estimated rMSE for $\mu = 0$ and $\sigma = 1$ for varying sample sizes</td>
<td>107</td>
</tr>
<tr>
<td>B.6</td>
<td>Estimated rMSE for $\mu = 0$ and $\sigma = 3.5$ for varying sample sizes</td>
<td>108</td>
</tr>
<tr>
<td>B.7</td>
<td>Estimated rMSE for $\mu = 2$ and $\sigma = 0.5$ for varying sample sizes</td>
<td>109</td>
</tr>
<tr>
<td>B.8</td>
<td>Estimated rMSE for $\mu = 2$ and $\sigma = 1$ for varying sample sizes</td>
<td>110</td>
</tr>
<tr>
<td>B.9</td>
<td>Estimated rMSE for $\mu = 2$ and $\sigma = 3.5$ for varying sample sizes</td>
<td>111</td>
</tr>
</tbody>
</table>
List of Tables

3.1 Summary statistics for the distributions of discrete sample arc length statistics ........................................... 13
3.2 Summary statistics for the distributions of continuous sample arc length statistics ...................................... 14
3.3 Estimated bias and rMSE for \( \mu \) ......................................................... 16
3.4 Estimated bias and rMSE for \( \sigma \) ......................................................... 18

4.1 Comparison of bootstrap results for OLS, MMM and ALR methods ......................................................... 38
4.2 Comparison of performance results for OLS, MMM and ALR methods .................................................... 39

5.1 \((h, k)\) combinations for the four-parameter kappa distribution ................................................................. 51
5.2 Comparison of bootstrap results for NLS and NALR methods for sample1 .................................................. 54
5.3 Comparison of performance results for NLS and AL methods for sample1 ................................................ 55
5.4 Comparison of bootstrap results for NLS and NALR methods for sample2 ................................................ 57
5.5 Comparison of performance results for NLS and AL methods for sample2 ................................................ 58

B.1 Estimated bias and rMSE for \( \mu=-2 \) and \( \sigma=0.5 \) ........................................................................ 101
B.2 Estimated bias and rMSE for \( \mu=-2 \) and \( \sigma=0.5 \) ........................................................................ 106
B.3 Estimated bias and rMSE for \( \mu=-2 \) and \( \sigma=1 \) ...................................................................... 112
B.4 Estimated bias and rMSE for \( \mu=-2 \) and \( \sigma=1 \) ...................................................................... 113
B.5 Estimated bias and rMSE for \( \mu=-2 \) and \( \sigma=3.5 \) ................................................................. 114
B.6 Estimated bias and rMSE for \( \mu=-2 \) and \( \sigma=3.5 \) ................................................................. 114
B.7 Estimated bias and rMSE for \( \mu=0 \) and \( \sigma=0.5 \) ....................................................................... 115
B.8 Estimated bias and rMSE for \( \mu=0 \) and \( \sigma=0.5 \) ....................................................................... 115
B.9 Estimated bias and rMSE for \( \mu=0 \) and \( \sigma=1 \) ....................................................................... 116
B.10 Estimated bias and rMSE for \( \mu=0 \) and \( \sigma=1 \) ...................................................................... 116
B.11 Estimated bias and rMSE for \( \mu=0 \) and \( \sigma=3.5 \) ................................................................. 117
B.12 Estimated bias and rMSE for $\mu=0$ and $\sigma=3.5$ . . . . . . . . . . . . . . . 117
B.13 Estimated bias and rMSE for $\mu=2$ and $\sigma=0.5$ . . . . . . . . . . . . . . . 118
B.14 Estimated bias and rMSE for $\mu=2$ and $\sigma=0.5$ . . . . . . . . . . . . . . . 118
B.15 Estimated bias and rMSE for $\mu=2$ and $\sigma=1$ . . . . . . . . . . . . . . . 119
B.16 Estimated bias and rMSE for $\mu=2$ and $\sigma=1$ . . . . . . . . . . . . . . . 119
B.17 Estimated bias and rMSE for $\mu=2$ and $\sigma=3.5$ . . . . . . . . . . . . . . . 120
B.18 Estimated bias and rMSE for $\mu=2$ and $\sigma=3.5$ . . . . . . . . . . . . . . . 120

C.1 Comparison of performance results for random forest and support vector
regression methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 124
Chapter 1

Introduction

Statisticians have always been interested in the development and refinement of procedures, i.e. those having greater statistical power, efficiency, maximum entropy, largest $r^2$, smallest variance, ..., and the list goes on. The ingredient providing the edge, of one technique over another, may for instance be quantified using measures from information theory, which simply states that the procedure contributing more “information” is preferred above the other. It is this prospecting for more information, that has developed into the subject of the current enquiry. Such developments have been heightened in the current age, where statisticians find themselves in a balancing game of theory and computation, as described by Efron and Hastie (2016) [25].

The approach here is therefore to use a mathematically intractable measure, namely the arc length, as pensieve by means of which musings on the positioning of classical Statistics within the broader field of Data Science, are provided. To this end, parameter estimation, goodness-of-fit, regression and non-linear regression will be revisited, and redefined in terms of the arc length of suitable statistical functions. Reflections along these lines are found in “Data Science: The End of Statistics” by Larry Wasserman\(^1\), and on the exercising of new-found cloud powers by Jonathan Rosenblatt\(^2\) which are brought about by high performance computing.

\(^1\)https://normaldeviate.wordpress.com/2013/04/13/data-science-the-end-of-statistics/
\(^2\)https://www.r-statistics.com/2013/07/analyzing-your-data-on-the-aws-cloud-with-r/
Chapter 1. Introduction

Holst and Rao (1980) [43] provides the description of the problem of random spacings in everyday English, when the lengths of a stick, broken at random, are discussed. This problem is however much older, and was considered by Pearson (1902) [68], when he studied the mean value of a spacing in response to a problem posed by Francis Galton. Many of the earlier approaches to the random division of an interval are surveyed by Moran (1947) [65], in a response to Greenwood (1946) [38]. Yet another review of the earlier contributions to random spacings was made by Pyke (1965 and 1972) [71, 72] and discussions found therein.

Arc lengths has been considered, and explicitly mentioned, by Rao (1976) [74], in the context of the distance between observations on the circle, i.e. spacings in the circular world, where the goodness-of-fit (GoF) and two-sample problem were treated. Tung and Rao (2012) [79] used U-statistics (see Hoeffding (1948) [41]) for developing U-statistics based on spacings. They proposed yet another test for uniformity on the circle in (2013) [80], by using Gini’s mean difference statistic for sample arc lengths. These circular sample arc lengths correspond to sample spacings on the real line. Since many applications of spacings are in directional statistics, or since the probability integral transform is often used in changing a problem on the real line into a problem on the circle, it is worth remarking the contributions by Mardia and Jupp (2009) [57], and Jammalamadaka and Sengupta (2001) [47]. In their work, the developments, and general theory of directional statistics is reviewed, and the theory for distributions wrapped around the circle is discussed, but is also generalised to other manifolds.

Cressie (1976) [17] generalised first order spacings to \( m \)-order spacings, and used the sum of their logarithms in testing for uniformity. For this, he also showed asymptotic normality under the null hypothesis. Holst (1979) [42] showed the asymptotic normality of functions of spacings, involving sums. The asymptotic theory of functions of spacings was further studied by Holst and Rao (1980, 1981) [43, 44], where they showed its application to the two-sample case. Here the non-parametric flavour of these tests are emphasised. Gatto and Rao (1999) [34] provide a saddlepoint approximation for an \( m \)-statistic conditional on another. A special case of this is that dependent spacings may be written in terms of independent and identically distributed (IID) random variables, conditioned on their sum. The sample arc length statistics proposed here, involve the summation of spacings, of order one, and the mapping of that into the range of the statistical function, i.e. creating a dependence structure.
Chapter 1. Introduction

Saddlepoint approximations were introduced into the statistical literature by Daniels (1954) [18], and is frequently encountered in the theory of spacings. Ma and Robinson (1999) [56] used saddlepoint approximations in deriving approximate distributions for the difference of order statistics. In other words, saddlepoint approximations were used in finding approximate distributions of spacings. The book by Butler (2007) [11] provides an overview of its use, and extensions to the multivariate case, as well as a discussion on how it may be used with other base distributions than the normal. Ghosh and Rao (1998) [35] used saddlepoint approximations in deriving approximate distributions for statistics of spacings involving small samples, and also tabled some values for distributions that can’t be derived explicitly.

Despite the rich literature on the theory of spacings, and the use of saddlepoint approximations, no exact or approximate distributions for the sample arc length statistics presented here, could be obtained. However, as mentioned above, the tabulation of critical values are often encountered for sample statistics involving spacings. An extended table for Rao’s Spacing Test is for instance given by Russell and Levitin (1995) [75]. Chen (2004) [13], used simulation when a test for a difference in scale between two populations, based on spacings, was proposed.

Since explicit solutions to arc length formulae are seldom obtained, the voyage of finding the most suitable candidate distribution for the problem, is abandoned from the outset. The point of departure, then, is approximation and simulation. This notion is supported by McElreath (2016) [58] in commenting on computational statistics, that a purely mathematical approach is anyways insufficient.

1.1 Motivation

The two questions posed by Larry Wasserman (2013) [84], provides the fuel for the work presented here:

1. Why do Statisticians find themselves left out of the Data Science discussion?

2. What can Statisticians do about it?

With regards to the first, assuming that computational capabilities are infinite, can Statisticians offer theoretically sound methodologies, which improve on those existing in the literature? When the day dawns that the disclaimer “These are computationally
intensive methods requiring supercomputing facilities...” can safely be left out of newly proposed work, what will the Statistician have to offer? Heeding the warning of Leonardo da Vinci quoted at the outset of this work, and echoed by Azzalini and Scarpa (2012) [7] (amongst others), a theoretical framework is proposed for the computationally intensive methods presented here.

Secondly, making use of parallel and high performance computing, the methods presented here, should be refined, and the available options, selected as to maximise their performance. These options include optimisation methodologies, and numerical analysis routines, such as numerical integration.

1.2 Objectives

- Develop a method for parameter estimation using arc lengths, namely the method of arc lengths.
- Develop a GoF test using arc lengths.
- Present a framework for arc length regression (ALR).
- Extend the ALR framework to the non-linear setting, namely non-linear arc length regression (NALR).
- Formulate the mathematical premises for arc length based techniques.

1.3 Contributions

The novel contributions this thesis sets out to make are:

- The development of an estimation method, that renders biased estimates, with smaller root-mean-square error (rMSE) and biases than existing methods.
- Determine conditions under which rejection regions for a newly proposed GoF test are larger than that of competitive tests, i.e. rendering a test with greater sensitivity.
• The presentation of a flexible linear regression technique, that utilises tuning parameters which yields larger $r^2$ values, and smaller sample divergences, using the same model specification as ordinary least squares (OLS).

• Propose a non-linear regression technique for fitting a sigmoidal function, under non-linear parameter constraints, as an alternative to non-linear least squares (NLS).

• Develop an R [73] package for enhancing the reproducibility of all methods proposed here.

1.4 Thesis Outline

• Chapter 2 provides the necessary background for developing the basic intuition for the arc length based techniques to follow in the consecutive chapters.

• Chapter 3 focuses on the classical estimation problem in Statistics. The method of arc lengths, along with the arc length test for GoF is presented.

• Chapter 4 extends the idea of a sequence of data values, to a set of dependent and independent observations. Here, arc length regression and the method of moments (regression) are introduced.

• Chapter 5 generalises the linear model assumption of Chapter 4 to the non-linear setting.

• Chapter 6 proposes the mathematical premises for the techniques presented in this thesis.

• Chapter 7 summarises the main findings of this thesis.

The set of appendices include:

• Appendix A describes the “alR” [54] package documentation for the functions used in this thesis.

• Appendix B provides additional simulation results for the estimation technique, applied to the normal distribution, introduced in Chapter 3.
• **Appendix C** includes comparative analyses from two popular machine learning techniques, for the data used in Chapter 4.

• **Appendix D** provides a list of the acronyms used throughout this thesis, as well as their associated definitions.

• **Appendix E** lists and defines the mathematical symbols used in this work, categorised according to the relevant chapter in which they appear.

• **Appendix F** lists the publications derived from, and that is associated with this work.

An index of some of the key terms used in this thesis, begin on page 129.
Chapter 2

Preliminaries

In this chapter, the mathematical background is reviewed in Section 2.1, required for the development of a basic intuition for the methods proposed in later chapters. Section 2.2 describes the implementation infrastructure required for bringing all this together.

2.1 Mathematical Background

Kent, Mardia and Rao (1979) [49] proved a characterisation of the uniform distribution on the circle. It is proposed that distributions may be characterised, not only on the circle, but also on the real line, by the lengths of arcs of statistical functions, over partitions of its domain. It makes for instance intuitive sense that the arc length of the probability density function (PDF), of a unimodal random variable, will be halved by the median, i.e. have equal lengths to the left and right of the fiftieth percentile. The formal definition of an arc length is now given in Definition 2.1.1.

**Definition 2.1.1.** Define the arc length of the function \( f(x, \theta) \), on the interval \([a, b]\) (on which it is continuous) as

\[
S_f^{[a,b]}(\theta) = \int_a^b \sqrt{1 + (f'(x, \theta))^2} \, dx
\]

(2.1.1)

where \( f'(\cdot) \) denotes the derivative of \( f(\cdot) \) with respect to \( x \), \( \theta \) is additional parameters of the function \( f(\cdot) \), and where \( a \leq b \) is in the domain of \( f(\cdot) \).

For the normal distribution, the PDF with mean \( \mu \in \mathbb{R} \) and standard deviation \( \sigma > 0 \)
is for instance given by

\[ f(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} \]

in the usual way, so that (2.1.1) becomes

\[
S_f^{[a,b]}(\mu, \sigma) = \int_a^b \sqrt{1 + \left(\frac{f'(x, \mu, \sigma)}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}\right)^2} \, dx \\
= \int_a^b \sqrt{1 + \left(-\frac{x-\mu}{\sigma^2} \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}\right)^2} \, dx \\
= \int_a^b \sqrt{1 + \frac{(x-\mu)^2}{2\pi\sigma^6} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}} \, dx. \quad (2.1.2)
\]

In Chapter 3 for instance, the discussion is concerned with the arc length of the PDF, so that the subscript may be dropped, i.e. \( S_f^{[a,b]}(\mu, \sigma) \) is used throughout.

The optimal choice of the interval \([a, b]\), remains an open question, and is viewed as tuning parameters. However, selection of an interval, covering the greater portion of the range of observed values, seems like a fair strategy. For example, \([a, b] = [q_{0.025}, q_{0.975}]\) results in an arc spanning the 0.025th to the 0.975th sample quantiles; thus excluding 5% of the data values. This is done, since the arc length over an infinite domain, would also be infinite.

2.2 Implementation Infrastructure

An “R” (see [73]) package “alR” (see Loots (2017) [54]) was developed for the methods proposed here. Code has been written in c++ as far as possible using “Rcpp” by Eddelbuettel and François (2011) [22], and Eddelbuettel (2013) [21] and “RcppArmadillo” by Eddelbuettel and Sanderson (2014) [23].

Because of the computational intensive nature of the methods proposed, the Lengau supercomputer from the CHPC\(^1\) was used, which is a Dell, Linux, homogeneous cluster, comprising Intel 5th generation CPUs. The amount of compute nodes used for a particular application will be noted where applicable. The job distribution was performed using task-pull parallelism from the “pbdMPI” package by Chen et al (2012) [12]).

\(^1\)http://www.chpc.ac.za/
Further technical requirements specific to a particular chapter will be provided where necessary.

### 2.3 Summary

The fundamental mathematical concepts for the development of the methods proposed in the following chapters were reviewed in Section 2.1. The technical specifications regarding their implementation and execution were presented in Section 2.2.

In Chapter 3, the first musings on the use of the arc length measure is presented for cross-sectional data, for estimation and goodness-of-fit problems.
Chapter 3

Estimation

In this chapter, the method of arc lengths applied to the estimation of the parameters of normally distributed variables, is presented, based on the arc length of the PDF. It is furthermore shown how arc lengths may be used for GoF problems (as the arc length test).

Here, the lengths of a broken stick aren’t considered explicitly, as by Holst and Rao (1980) [43] (as mentioned in Chapter 1) but rather the lengths of a PDF, “broken” at random, i.e. the arc lengths of a density function over sub-domains of its support. After providing a basic definition and describing the methods utilised here in Section 3.1, a sample statistic is formulated in terms of spacings, and functions thereof, in Section 3.2. Apart from the review on spacings given in Chapter 1, it is noted here that spacings have also been used in the construction of GoF tests, by Ghosh and Jammalamadaka (2001) [36], Ekström (2008) [26], and Mirakhmedov and Jammalamadaka (2013) [64].

The method of arc lengths proposed here is similar to that of the minimum distance estimators (or maximum goodness-of-fit estimators) as described by Wolfowitz (1953) [86], Kac, Kiefer and Wolfowitz (1955) [48], Pollard (1980) [69], and more recently by Luceño (2006) [55], in that:

1. it may be used in parameter estimation (Section 3.3), and
2. it may be used as a measure of GoF for which critical values are simulated via a parametric bootstrap (Section 3.4).

Two special distances that will receive attention here, for comparative purposes, are Cramér-von Mises (CvM), which may be regarded as a special case of Anderson-Darling
(AD) (see Anderson and Darling (1952) [4]).

### 3.1 Implementation Infrastructure

In addition to the specifications set out in Chapter 2, the following applies to this chapter.

The violin plots in Figures 3.1 and 3.2 were constructed using the “vioplot” package (see Adler (2005) [2]), based on Hintze and Nelson (1998) [40]; and the heatmaps in Figures 3.4 and 3.6 using “heatmap.2” from “gplots” (see Warnes et al (2016) [83]).

The construction of these heatmaps required the Lengau supercomputer from the CHPC. For Figure 3.4, 6 compute nodes with 24 cores and 128 GiB memory each were utilised, but was down-scaled to 5 compute nodes with 24 cores each for Figure 3.6.

Parameter estimation and GoF were respectively carried out using the packages “fitdistrplus” by Delignette and Dutang (2015) [19] and “goftest” by Faraway et al (2015) [29]. The random samples from the skew normal distribution and the power exponential distribution were respectively simulated using “sn” by Azzalini (2016) [6] and “normalp” by Mineo (2014) [62].

### 3.2 Sample Arc Length Statistics

Suppose that \(x_1, \ldots, x_n\) is an observed sample from a distribution, having PDF \(f(x)\), with corresponding ordered values \(x_{(1)}, \ldots, x_{(n)}\). Suppose further that \([a, b]\) is some interval in the domain of \(f\), such that \(a \leq x_{(1)}^* \leq \ldots \leq x_{(m)}^* \leq b\). Here \(x_{(i)}^*, i = 1, \ldots, m\) denote the \(m\) ordered observations in the interval \([a, b]\). (2.1.2) is then estimated by a finite sum of lengths of straight lines, as

\[
P_{[a,b]} = \sum_{i=2}^{m} \sqrt{\left(x_{(i)}^* - x_{(i-1)}^*\right)^2 + \left(\hat{f}\left(x_{(i)}^*ight) - \hat{f}\left(x_{(i-1)}^*ight)\right)^2},
\]

where \(\hat{f}()\) is the kernel density estimate (KDE) of \(f()\), based on the Gaussian kernel, and is given by

\[
\hat{f}(x) = \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}hn} e^{-\frac{(x-x_i)^2}{2hn^2}},
\]

and where

\[
h = s_x \left(\frac{4}{3n}\right)^{1/5},
\]
is the smoothing parameter suggested by Silverman (1986) [77]. The choice of the kernel and smoothing parameter correspond to the hypothesis, that the observed data is normally distributed.

Since the KDE has already been mentioned, and is a PDF in its own right, another estimator resulting from this, is

\[ 2f^{[a,b]} = \int_a^b \sqrt{1 + \left( \hat{f}'(x) \right)^2} \, dx, \]  

which is the continuous analogue of (3.2.1).

Since explicit distributions of the sample arc length statistics aren’t derived explicitly, 1,000 samples of size 1,000 were simulated from the N (2, 3.5) distribution, and (3.2.1) and (3.2.2) calculated, in providing an empirical distribution for each. The violin plots for the measurements generated by (3.2.1) and (3.2.2), are given in Figures 3.1 and 3.2, for three interval choices:

(a) \([a, b] = [q_{0.01}, q_{0.99}] = [-6.1422175591, 10.1422175591],\)

(b) \([a, b] = [q_{0.05}, q_{0.95}] = [-3.7569876943, 7.7569876943],\) and

(c) \([a, b] = [q_{0.10}, q_{0.90}] = [-2.4854304794, 6.4854304794].\)

The final row in Table 3.1 and 3.2 indicates the quantile at which the following “theoretical” arc lengths (2.1.2) were respectively observed (sij, i = 1, 2, and j = a, b, c refer to the estimators and interval choices respectively):

(a) \(S^{[-6.1422175591, 10.1422175591]}(2, 3.5) = 16.2860589511,\)

(b) \(S^{[-3.7569876943, 7.7569876943]}(2, 3.5) = 11.5153831966,\) and

(c) \(S^{[-2.4854304794, 6.4854304794]}(2, 3.5) = 8.9719304133.\)

Note that for the discrete sample statistic, given in (3.2.1), the “theoretical” value was only observed in the range of the empirical distribution once the segment over which the arc length was computed included less tail observations.
Figure 3.1: Violin plots for the empirical distributions of discrete arc length statistics: \( f_{[-6.1422175591,10.1422175591]}(2,3.5) \), \( f_{[-3.7569876943,7.7569876943]}(2,3.5) \), and \( f_{[-2.4854304794,6.4854304794]}(2,3.5) \).

<table>
<thead>
<tr>
<th></th>
<th>s1a</th>
<th>s1b</th>
<th>s1c</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min</td>
<td>15.16233</td>
<td>11.16418</td>
<td>8.76773</td>
</tr>
<tr>
<td>Q1</td>
<td>15.96189</td>
<td>11.42711</td>
<td>8.91885</td>
</tr>
<tr>
<td>Mean</td>
<td>16.04576</td>
<td>11.44779</td>
<td>8.93275</td>
</tr>
<tr>
<td>Median</td>
<td>16.07521</td>
<td>11.45979</td>
<td>8.93825</td>
</tr>
<tr>
<td>Q3</td>
<td>16.16260</td>
<td>11.48252</td>
<td>8.95326</td>
</tr>
<tr>
<td>Max</td>
<td>16.27826</td>
<td>11.51517</td>
<td>8.97279</td>
</tr>
<tr>
<td>F(S)</td>
<td>1.00000</td>
<td>1.00000</td>
<td>0.99600</td>
</tr>
</tbody>
</table>

Table 3.1: Summary statistics for the distributions of discrete sample arc length statistics.
Figure 3.2: Violin plots for the empirical distributions of continuous arc length statistics: $2\int_{-6.1422175591}^{10.1422175591}(2, 3.5)$, $2\int_{-3.7569876943}^{7.7569876943}(2, 3.5)$, and $2\int_{-2.4854304794}^{6.4854304794}(2, 3.5)$.

<table>
<thead>
<tr>
<th></th>
<th>s2a</th>
<th>s2b</th>
<th>s2c</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min</td>
<td>16.28563</td>
<td>11.51479</td>
<td>8.97138</td>
</tr>
<tr>
<td>Q1</td>
<td>16.28588</td>
<td>11.51518</td>
<td>8.97172</td>
</tr>
<tr>
<td>Mean</td>
<td>16.28598</td>
<td>11.51529</td>
<td>8.97184</td>
</tr>
<tr>
<td>Median</td>
<td>16.28597</td>
<td>11.51529</td>
<td>8.97184</td>
</tr>
<tr>
<td>Q3</td>
<td>16.28607</td>
<td>11.51539</td>
<td>8.97196</td>
</tr>
<tr>
<td>Max</td>
<td>16.28652</td>
<td>11.51577</td>
<td>8.97261</td>
</tr>
<tr>
<td>F(S)</td>
<td>0.71700</td>
<td>0.73700</td>
<td>0.69300</td>
</tr>
</tbody>
</table>

Table 3.2: Summary statistics for the distributions of continuous sample arc length statistics.

From Figure 3.2, and the summary statistics in Table 3.2, the robustness of the continuous sample arc length statistic is evident, given the small variation in the observed values.
3.3 Parameter Estimation

Generally, a set of observations is tested under the null hypothesis of being generated by a normal distribution, for which the parameters must be estimated, i.e. let $X_1, \ldots, X_n \sim N(\mu, \sigma)$ be IID, with $\mu$ and $\sigma$ unknown. The $(\hat{\mu}, \hat{\sigma})$ combination that satisfies

$$S_{[a,b]}(\mu, \sigma) = f_{[a,b]}$$

where $f_{[a]}$ is one of the sample arc length statistics defined in Section 3.2. This leads to the following objective function, based on the Euclidean distance:

$$\sqrt{(S_{[a,b]}(\mu, \sigma) - f_{[a,b]})^2}.$$ (3.3.1)

$(\hat{\mu}, \hat{\sigma})$ is therefore the $(\mu, \sigma)$ pair that simultaneously minimises this objective function. In this case, simplification of (3.3.1) results in the absolute difference, however, the degree of fit may be controlled by the inclusion of additional subintervals, and expansion of the Euclidean distance in the usual fashion. This however complicates the matter of choosing optimal tuning parameters, i.e. selecting integration intervals for which this Euclidean distance is minimised.

Tables 3.3 and 3.4 summarise the estimated bias and rMSE values (in ascending order of rMSE) for the median values of the parameters estimated from 1,000 samples of size 1,000, from a $N(2, 3.5)$ distribution. The method of arc lengths (resulting from the two proposed sample arc length statistics given in (3.2.1) and (3.2.2)), maximum-likelihood (ML), CvM, and AD estimation techniques are compared. For each of the proposed sample arc length statistics, five variants were included depending on the interval $[a, b]$, where the two end points are specified in terms of the sample quantiles:

(a) $[0.01, 0.99],$
(b) $[0.025, 0.975],$
(c) $[0.05, 0.95],$
(d) $[0.075, 0.925]$, and
(e) $[0.1, 0.9].$
<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00735</td>
<td>0.07379</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00819</td>
<td>0.07610</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00804</td>
<td>0.07935</td>
</tr>
<tr>
<td>s1b</td>
<td>0.08767</td>
<td>0.12631</td>
</tr>
<tr>
<td>s1c</td>
<td>0.14413</td>
<td>0.17853</td>
</tr>
<tr>
<td>s1a</td>
<td>0.15275</td>
<td>0.18645</td>
</tr>
<tr>
<td>s2e</td>
<td>0.20622</td>
<td>0.21606</td>
</tr>
<tr>
<td>s2a</td>
<td>0.22728</td>
<td>0.23129</td>
</tr>
<tr>
<td>s2d</td>
<td>0.22842</td>
<td>0.23211</td>
</tr>
<tr>
<td>s2c</td>
<td>0.23264</td>
<td>0.23988</td>
</tr>
<tr>
<td>s2b</td>
<td>0.24070</td>
<td>0.24381</td>
</tr>
<tr>
<td>s1e</td>
<td>0.26803</td>
<td>0.26803</td>
</tr>
<tr>
<td>s1d</td>
<td>0.27966</td>
<td>0.28752</td>
</tr>
</tbody>
</table>

Table 3.3: Estimated bias and rMSE for $\mu$.

Using the method of arc lengths with $\int_{[0.01,0.99]}$ yielded the smallest bias and rMSE values for simultaneously estimating $\mu$ and $\sigma$. All sample arc length statistics performed worse than the maximum-likelihood estimation (MLE), AD and CvM methods for estimating $\mu$. The same holds true for the estimation of $\sigma$, accept for $\int_{[0.01,0.99]}$ and CvM, for which the ordering are interchanged in certain cases (see Appendix B). In Figure 3.3 these results are reproduced using only $\int_{[0.01,0.99]}$, for varying sample sizes.

From Figure 3.3, it is clear that the rMSE values for the method of arc lengths with $\int_{[0.01,0.99]}$ decrease for increasing sample sizes. For estimating $\sigma$, this corresponds well to the values obtained for the CvM method, but are not as strong as that obtained for the MLE and AD methods.

Appendix B provides simulation results for a number of $\mu$ and $\sigma$ choices, and for varying sample sizes, with similar results given here.
Chapter 3. Estimation

Figure 3.3: Estimated rMSE for μ and σ for varying sample sizes.
Table 3.4: Estimated bias and rMSE for $\sigma$.

<table>
<thead>
<tr>
<th>Method</th>
<th>Bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00256</td>
<td>0.05460</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00118</td>
<td>0.05638</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00956</td>
<td>0.06043</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00705</td>
<td>0.06375</td>
</tr>
<tr>
<td>s2a</td>
<td>0.06186</td>
<td>0.08131</td>
</tr>
<tr>
<td>s2b</td>
<td>0.06544</td>
<td>0.08554</td>
</tr>
<tr>
<td>s2c</td>
<td>0.07228</td>
<td>0.09172</td>
</tr>
<tr>
<td>s2d</td>
<td>0.07086</td>
<td>0.09981</td>
</tr>
<tr>
<td>s2e</td>
<td>0.07402</td>
<td>0.10673</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.13979</td>
<td>0.13993</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.35670</td>
<td>0.35670</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.56492</td>
<td>0.56492</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.75129</td>
<td>0.75129</td>
</tr>
</tbody>
</table>

3.4 Goodness-Of-Fit

After obtaining estimates $\hat{\mu}$ and $\hat{\sigma}$ of $\mu$ and $\sigma$, from a sample $\mathbf{x} = x_1, \ldots, x_n$, a GoF procedure may be setup using $2\int$. The procedure for the arc length test is as follows:

1. Under the null hypothesis: $H_0: \mathbf{x} \sim N(\hat{\mu}, \hat{\sigma})$.

2. Calculate $2\int$ from $\mathbf{x}$ as a test statistic.

3. Simulate $m$ samples of size $n$ from the $N(\hat{\mu}, \hat{\sigma})$ distribution.

4. Calculate $2\int$ in each case, i.e. a parametric bootstrap.

5. A two-sided $p$-value may then be obtained from this bootstrap distribution in the usual fashion.

The choice of the interval $[a, b]$ yield tests of varying strengths, and a number of these will be investigated. This class of GoF procedures is now compared to the CvM and AD methods in detecting departures from normality, first against skew alternatives, and

\[^1\]Due to the discrete nature of $1\int$, it is not particularly useful in a GoF setting.
secondly against alternatives with mesokurtic deviations. For the arc length test, the parameters are estimated using $f_1$ from (3.2.1), and $f_2$ from (3.2.2), and for the CvM and AD tests, the CvM and AD estimation methods are respectively used. In other words, four GoF procedures will be compared.

### 3.4.1 Skewness Considerations

The skew normal distribution proposed by Azzalini (1985) [5] has PDF

$$f(x, \mu, \sigma, \alpha) = \frac{1}{\pi \sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \int_{-\infty}^{x-\mu} e^{-\frac{t^2}{2}} dt$$

where $x \in \mathbb{R}$, and has location parameter $\mu \in \mathbb{R}$, scale parameter $\sigma > 0$, and shape parameter $\alpha \in \mathbb{R}$, and includes the normal distribution as a special case, when the “skewness” parameter $\alpha = 0$. It is desired to test how sensitive the proposed arc length method is in detecting skewed departures from normality. Random samples of size 1,000 were simulated from the skew normal distribution, for the skewness parameter in the interval $[-3, 3]$, and in each case:

1. it was assumed that the data was generated by a normal distribution,
2. $(\hat{\mu}, \hat{\sigma})$ were estimated from the sample,
3. the $p$-values for the respective GoF methods were recorded, and
4. the entire experiment was repeated 100 times, and the median $p$-value at each skewness level recorded, in stabilising results.

The axis labels in the heatmaps indicate the range of the sub-domains over which the arc lengths were calculated, with equal distances from each endpoint. E.g. $f_{[0.03, 0.97]}^{[0.03, 0.97]}$ will be included with the $0.97 - 0.03 = 0.94$ label.
Chapter 3. Estimation

Figure 3.4: Heatmaps of $p$-values for GoF using arc lengths (skewness considerations).

Significance levels often range between 0.01 and 0.05 and the $p$-values corresponding to this are indicated by the yellow regions in the heatmaps, while $p$-values smaller than 0.01 are coloured red. Using the discrete sample arc length statistic $\int_{[q_{0.03},q_{0.97}]}$ seems to be the optimal choice at a 5% level of significance, since any larger intervals will never reject the null hypothesis, while smaller intervals, will almost always lead to its rejection. From the green heatmap represented by $\int_{q_{0.1}}$, it is clear that although the parametric bootstrap distribution under the null hypothesis, is simulated using this continuous sample arc length statistic, it never leads to its rejection.
From Figure 3.5 it is clear that the CvM test never leads to the rejection of the null hypothesis for the skewness parameter in the interval $[-3, 3]$. The arc length test (using $\int_{q_{0.03}}^{q_{0.97}}$) rejects the null hypothesis for values smaller than -2.1, (ignoring the jump at -2.05) and then again for values greater than 1.85 (inclusive). The AD test stops rejecting the null hypothesis at roughly the same point as the arc length test, but only starts rejecting again at 2.1. Thus although the arc length test rejects for a larger range of positively skewed alternatives, the AD test does so with greater decisiveness (i.e. a larger peak).

Figure 3.5: GoF comparisons for skew alternatives.
3.4.2 Mesokurtic Considerations

The exponential power distribution is a generalisation of the normal distribution, for which a third parameter $p$ is introduced in rendering mesokurtic deviances. It has PDF

$$f(x, \mu, \sigma, p) = \frac{1}{2p^\frac{1}{p} \Gamma(1 + \frac{1}{p})\sigma} e^{-\frac{|x-\mu|^p}{p \sigma^p}}$$

for $x \in \mathbb{R}$, location parameter $\mu \in \mathbb{R}$, scale parameter $\sigma > 0$, and shape parameter $p > 0$. For a history on this distribution, and the various names it goes by, see Mineo et al (2005) [63]. This distribution has as special cases the Laplace distribution for $p = 1$ (leptokurtic), normal distribution for $p = 2$, and tends to the uniform (platykurtic), as $p \to \infty$.

Random samples of size 1,000 were simulated from the exponential power distribution, for $p$ in the interval [1, 5], with the remainder of the experiment following that outlined in Section 3.4.1.
Figure 3.6: Heatmaps of p-values for GoF using arc lengths (mesokurtic deviations).

As with the skew alternatives considered in Figure 3.4, using the discrete sample arc length statistic \( 1f_{[0.03,0.97]} \) seems to be the optimal choice at a 5% level of significance, since any larger intervals will decrease the rejection region, while smaller intervals increase the Type I error. The case for the continuous sample arc length statistic \( 2f \), is similar to that described for skew alternatives.
From Figure 3.7 it is clear that the CvM test never leads to the rejection of the null hypothesis for \( p \in [1, 5] \). The arc length test (using \( 1 f^{[0.03,0.07]} \)) rejects the null hypothesis for values larger than 2.04, but not for any leptokurtic alternatives. The AD test stops rejecting the null hypothesis at \( p = 1.28 \), but only starts rejecting again at 3.96. Thus the AD test is more sensitive to leptokurtic alternatives, while the arc length test has greater sensitivity against platykurtic alternatives.

### 3.5 Summary

The method of arc lengths, for parameter estimation, was introduced here, and it was shown that although it being consistent, generally leads to parameter estimates with greater bias and rMSE than ML, CvM, and AD methods. The discrete arc length sample statistic, \( 1 f^{[0.01,0.99]} \) proved to be particularly effective for estimating \( \sigma \), and offers results...
comparable to that obtained by the CvM method.

The arc length GoF test is constructed using the continuous sample arc length statistic $2 \int$, for which a parametric bootstrap yields a null distribution. A GoF test constructed with $2 \int_{0.03}^{0.97}$, with parameters estimated using $1 \int_{0.03}^{0.97}$ yielded the most powerful tests. It was shown that the arc length test is more powerful against positively skewed and platykurtic alternatives than both the CvM and AD tests.

In the following chapter, the single sequence of data considered here, is extended to sets of dependent and independent observations, as the focus shifts to linear regression.
Chapter 4

Arc Length Regression

Chapter 3, considered a single sequence of data, and discussed parameter estimation along with GoF. In this chapter, two related sets of data are considered, one being a dependent outcome of the other.

The geometric interpretation of OLS, in a regression context, is the process by which the set of coefficients are sought, that minimises the sum of the vertical distances between a data point, and the regression line assumed to describe the data. For a historical review, see Seal (1967) [76]. Although numerous extensions and generalisations of the basic regression model have been proposed, a particular variation, having a geometric interpretation, involving distances, are worth mentioning.

Orthogonal regression seeks to minimise the sum of the orthogonal distances from a point to the line, and not the vertical distance (see Coolidge (1913) [16] and Minda and Phelps (2008) [61]). This is a special case of what is now referred to as Deming regression (see Adcock (1878) [1], Kummell (1879) [50], Deming (1943) [20]), a type of errors-in-variables model, obtained when equality of the errors in the dependent and independent variables are assumed. This on its part, is again a special case of total least squares (TLS), for which no general solution exists (see Golub and Van Loan(1980) [37]).

The importance of geometric- as opposed to algebraic methods, in recent applications, are emphasised by Ahn (2008) [3], Liu and Wang (2008) [53], and Chernov (2011) [14]. It makes sense that a geometrical approach yields beneficial results in fitting curves to objects in space, but would there be any advantage in applying such techniques directly to PDF’s? After some preliminaries in Section 4.1, the framework for addressing this question is developed in Sections 4.2 and 4.3.
A problem in biology is then discussed as an application in Section 4.5, and the findings summarised in Section 4.6.

4.1 Introduction

Suppose that \( y \) is a vector of \( n \) dependent observations, and that \( X = (x_1, \ldots, x_p) \) is a set of \( p \) independent variables (where a column may or may not be included for an intercept). In OLS regression the following is assumed:

\[
y = X\beta + \epsilon
\]

with \( \beta \) the set of coefficients to be determined, and \( \epsilon \) a vector of stochastic error terms with \( E[\epsilon] = 0 \), and \( \text{var}(\epsilon) = \sigma^2I_n \), where \( I_n \) is the identity matrix. The stochastic mechanism by which modelling takes place, is then through

\[
E[y] = E[X\beta + \epsilon] = X\beta, \quad \text{and}
\]

\[
\text{var}(y) = \text{var}(X\beta + \epsilon) = \text{var}(\epsilon) = \sigma^2I_n,
\]

since \( X\beta \) is assumed fixed. The OLS solutions \( \hat{\beta} \) are unbiased, having minimum variance, and if the further assumption is made that

\[
\epsilon \sim N(0, \sigma I_n)
\]

i.e. following a normal distribution, then the coefficient estimates \( \hat{\beta} \) are given by the ML solutions with known distributions.

A closer look at this model shows that an equivalent formulation is to assume that

\[
\gamma \sim N(X\beta, \sigma I_n)
\]

with \( \gamma \), so that the model may be rewritten as

\[
y = \gamma
\]
Chapter 4. Arc Length Regression

with

\[ E[y] = E[\gamma] = X \beta, \text{ and} \]

\[ \text{var}(y) = \text{var}(\gamma) = \sigma^2 I_n. \]

Thus, instead of modelling the residuals, the dependent variable may be modelled along with the residuals, seeking coefficient estimates \( \hat{\beta} \) that will yield a characteristic that is equal for both the \( y \) and \( \gamma \) observations.

The assumption of normality is also redundant in this case, and a more general, non-parametric KDE, with Gaussian kernel, will be assumed for \( y \) and \( \gamma \). Thus

\[
\hat{f}_y(y) = \frac{1}{nh_y} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left( \frac{y - y_i}{h_y} \right)^2},
\]

with

\[ h_y = s_y \left( \frac{4}{3n} \right)^{1/5}, \]

being the smoothing parameter suggested by Silverman (1986) [77]. Similarly, the KDE for \( \gamma \) is given by

\[
\hat{f}_\gamma(\gamma) = \frac{1}{nh_\gamma} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left( \frac{\gamma - \gamma_i}{h_\gamma} \right)^2}
\]

\[ = \frac{1}{nh_\gamma} \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2} \left( \frac{\gamma - \sum_{j=1}^{p} x_{ij}\beta_j}{h_\gamma} \right)^2}, \]

with

\[ h_\gamma = s_\gamma \left( \frac{4}{3n} \right)^{1/5}. \]

4.2 Equating Moments

Non-parametric regression using kernel density estimators has long been proposed by Nadaraya (1964) [66], and Watson (1964) [85], and a general framework for moment

© University of Pretoria
matching by Song et al (2008) [78]. Here, it is proposed that the moments of $y$ and $\gamma$ be matched in order to solve for $\beta$. Thus for $\nu \in \mathbb{N}$, it follows from (4.1.1) that

$$
E_y[y^{\nu}] = \int_{-\infty}^{\infty} y^{\nu} \hat{f}_y(y) dy
$$

$$
= \int_{-\infty}^{\infty} \frac{1}{n} \sum_{i=1}^{n} y^{\nu} \frac{1}{\sqrt{2\pi h_y}} e^{-\frac{1}{2} \left( \frac{y-y_i}{h_y} \right)^2} dy
$$

$$
= \frac{1}{n} \sum_{i=1}^{n} \int_{-\infty}^{\infty} y^{\nu} \frac{1}{\sqrt{2\pi h_y}} e^{-\frac{1}{2} \left( \frac{y-y_i}{h_y} \right)^2} dy
$$

$$
= \frac{1}{n} \sum_{i=1}^{n} m'_\nu(y_i, h_y),
$$

where $m'_\nu(y_i, h_y)$ denotes the raw moments of a $\mathcal{N}(y_i, h_y)$ variable. In particular

$$
E_y[y] = \frac{1}{n} \sum_{i=1}^{n} y_i = \bar{y}
$$

$$
E_y[y^2] = \frac{1}{n} \sum_{i=1}^{n} y_i^2 + h_y^2
$$

$$
E_y[y^3] = \frac{1}{n} \sum_{i=1}^{n} y_i^3 + 3y_i h_y^2,
$$

and so on. Similarly, from (4.1.2), it follows that

$$
E_\gamma[\gamma^{\nu}] = \int_{-\infty}^{\infty} \gamma^{\nu} \hat{f}_\gamma(\gamma) d\gamma
$$

$$
= \int_{-\infty}^{\infty} \frac{1}{n} \sum_{i=1}^{n} \gamma^{\nu} \frac{1}{\sqrt{2\pi h_\gamma}} e^{-\frac{1}{2} \left( \gamma - \frac{\sum_{j=1}^{p} x_{ij} \beta_j}{h_\gamma} \right)^2} d\gamma
$$

$$
= \frac{1}{n} \sum_{i=1}^{n} \int_{-\infty}^{\infty} \gamma^{\nu} \frac{1}{\sqrt{2\pi h_\gamma}} e^{-\frac{1}{2} \left( \gamma - \frac{\sum_{j=1}^{p} x_{ij} \beta_j}{h_\gamma} \right)^2} d\gamma
$$

$$
= \frac{1}{n} \sum_{i=1}^{n} m'_\nu(\sum_{j=1}^{p} x_{ij} \beta_j, h_\gamma),
$$

© University of Pretoria
where \( m'_\nu \left( \sum_{j=1}^{p} x_{ij}\beta_j, h_\gamma \right) \) denotes the raw moments of a \( N \left( \sum_{j=1}^{p} x_{ij}\beta_j, h_\gamma \right) \) variable. In particular

\[
E_\gamma[\gamma] = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{p} x_{ij}\beta_j
\]

\[
E_\gamma[\gamma^2] = \frac{1}{n} \sum_{i=1}^{n} \left( \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + h_\gamma^2
\]

\[
E_\gamma[\gamma^3] = \frac{1}{n} \sum_{i=1}^{n} \left( \sum_{j=1}^{p} x_{ij}\beta_j \right)^3 + 3 \left( \sum_{j=1}^{p} x_{ij}\beta_j \right) h_\gamma^2,
\]

and so on.

An objective function may then be constructed as the minimum squared Euclidean distance between the first \( p \) moments of the kernel densities of \( y \) and \( \gamma \), i.e.

\[
\hat{\beta} = \min_{\beta} \sum_{\nu=1}^{p} (E_y[y^\nu] - E_\gamma[\gamma^\nu])^2.
\] (4.2.1)

For instance, for a model having an intercept term, and two independent variables, the following three equations are implicitly solved

\[ E_y[y] = E_\gamma[\gamma] \]

\[ \therefore \sum_{i=1}^{n} y_i = \sum_{i=1}^{n} \sum_{j=1}^{3} x_{ij}\beta_j \] (4.2.2)

\[ E_y[y^2] = E_\gamma[\gamma^2] \]

\[ \therefore \sum_{i=1}^{n} [y_i^2 + h_y^2] = \sum_{i=1}^{n} \left[ \left( \sum_{j=1}^{3} x_{ij}\beta_j \right)^2 + h_\gamma^2 \right] \] (4.2.3)

\[ E_y[y^3] = E_\gamma[\gamma^3] \]

\[ \therefore \sum_{i=1}^{n} [y_i^3 + 3y_i h_y^2] = \sum_{i=1}^{n} \left[ \left( \sum_{j=1}^{3} x_{ij}\beta_j \right)^3 + 3 \left( \sum_{j=1}^{3} x_{ij}\beta_j \right) h_\gamma^2 \right]. \] (4.2.4)

The left-hand-sides of these equations are all values obtained directly from the data, but the right-hand-sides are non-linear functions in the three unknowns \( \beta_1, \beta_2, \) and \( \beta_3 \). Note that although a Gaussian kernel is assumed for the KDE, it is not required to actually construct this in obtaining estimates for the model.
4.3 Arc Length Regression

Instead of finding the set of $\hat{\beta}$ values that yield

$$E[y] = E[\gamma]$$

(as in the OLS case), observe the arc lengths of the two kernel density functions, (4.1.1) and (4.1.2), on the interval $[a, b]$, assumed to be in both their domains

$$S_y^{[a,b]} = \int_a^b \sqrt{1 + \left( \hat{f}_y'(y) \right)^2} dy$$

$$S_\gamma^{[a,b]}(\beta) = \int_a^b \sqrt{1 + \left( \hat{f}_\gamma'(\gamma) \right)^2} d\gamma.$$  

An objective function may be constructed by partitioning the interval $[a, b]$ into $p$ sub-intervals, i.e. let $a = i_1 < \ldots < i_{p+1} = b$, so that $\hat{\beta}$ is given by the set of $\beta$ values that simultaneously minimises the $p$ equations

$$S_{y}^{[i_j,i_{j+1}]} = S_\gamma^{[i_j,i_{j+1}]}(\beta)$$

$$\vdots$$

$$S_{y}^{[i_p,i_{p+1}]} = S_\gamma^{[i_p,i_{p+1}]}(\beta),$$

or equivalently, that minimises the squared Euclidean distance between the two sets of $p$ arc lengths:

$$\hat{\beta} = \min_\beta \sum_{j=1}^{p} \left( S_{y}^{[i_j,i_{j+1}]} - S_\gamma^{[i_j,i_{j+1}]}(\beta) \right)^2.$$  

4.4 Implementation Infrastructure

In order to obtain solutions to (4.2.1) and (4.3.4) for the moment matching method (MMM) and ALR methods respectively, the Nelder-Mead simplex method by Nelder and Mead (1965) [67] from the “optim” function in “R” was used, with initial values resulting from OLS.

For calculation of standard errors, confidence intervals, $p$-values and biases, bootstrap resampling proposed by Efron (1982) [24] was implemented. These methods are computationally intensive, and for ALR, supercomputing facilities were required. For
this, the Lengau super computer from the CHPC was used, with 90 compute nodes with 24 cores and 128 GiB memory each.

### 4.5 Application

Hussey et al (2017) [46] were interested in calculating the percentage variation explained in expression data captured in “Raw_FPKM” signals by two histone modifications, “H3K4me3” and “H3K27me3”. Statistically speaking, this is a model with a single dependent variable, two independent variables, and an intercept. The data is summarised below:

<table>
<thead>
<tr>
<th>Gene_ID</th>
<th>Raw_FPKM</th>
<th>H3K4me3_signal_bin25</th>
<th>H3K27me3_signal_bin21</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 Eucgr.A00043</td>
<td>0</td>
<td>10.3002</td>
<td>25.5719</td>
</tr>
<tr>
<td>3 Eucgr.A00047</td>
<td>0</td>
<td>-11.8096</td>
<td>18.5719</td>
</tr>
<tr>
<td>4 Eucgr.A00088</td>
<td>0</td>
<td>-16.9475</td>
<td>-3.9692</td>
</tr>
<tr>
<td>5 Eucgr.A00090</td>
<td>0</td>
<td>-9.8643</td>
<td>-8.3181</td>
</tr>
<tr>
<td>6 Eucgr.A00098</td>
<td>0</td>
<td>1.6385</td>
<td>47.6019</td>
</tr>
</tbody>
</table>

A 60%–40% split of the data yielded a training set (of 21427 observations) and testing set (of 14284 observations) respectively. The Yeo-Johnson transformation, using the “car” package by Fox and Weisberg (2011) [33], has been applied to the “Raw_FPKM” signal, with optimal choice \( \lambda = -0.503013133065168 \). An OLS model has been fitted to the transformed variable, and the results reproduced here.

```r
Call: lm(formula = FPKM_lambda ~ H3K4me3_signal_bin25 + H3K27me3_signal_bin21, data = training)

Residuals:
     Min      1Q  Median       3Q      Max
-1.776  -0.329  -0.109   0.267   1.899

Coefficients:              Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.3629992  0.0042044   86.3  <2e-16
H3K4me3_signal_bin25  0.0046198  0.0000334  138.4  <2e-16
```

© University of Pretoria
The fitted KDE's, using Gaussian kernels, with Silverman’s bandwidth estimator, to the transformed “Raw_FPKM”, fitted model resulting from OLS estimates, and these results applied to the testing set, along with the respective bandwidth estimators, are illustrated in Figure 4.1. The solid (red) curves represent the KDE’s for the actual, and the blue curves the KDE’s for the estimated signals. For the training set $r^2 = 0.50266$, and for the testing set $r^2 = 0.49657$.

(a) KDE’s for transformed FPKM signal and the OLS fit thereof, with bandwidths 0.09333 and 0.06617 respectively (training set).

(b) KDE’s for transformed FPKM signal and the OLS fit thereof, with bandwidths 0.10109 and 0.07233 respectively (testing set).

**Figure 4.1:** Plots for transformed expression data resulting from OLS.

### 4.5.1 Moment Matching

Solving for the three coefficients above using MMM set out in Section 4.2, requires finding solutions to (4.2.2) – (4.2.4). For this, the “mmKDEboot” function from the
“alR” package is used in solving for \( \beta \). Here, bootstrapping is employed for estimating standard errors, confidence intervals, biases and \( p \)-values for the estimated coefficients. The results now follow.

<table>
<thead>
<tr>
<th>Residuals:</th>
<th>Min.</th>
<th>1st Qu.</th>
<th>Median</th>
<th>3rd Qu.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-3.925800</td>
<td>-0.328780</td>
<td>0.043892</td>
<td>0.351300</td>
<td>3.527000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KDE Moments:</th>
<th>LHS</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.66191</td>
<td>0.66191</td>
</tr>
<tr>
<td>2</td>
<td>0.86605</td>
<td>0.86605</td>
</tr>
<tr>
<td>3</td>
<td>1.24962</td>
<td>1.24962</td>
</tr>
<tr>
<td>BW</td>
<td>0.09333</td>
<td>0.09333</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Value of objective function:</th>
<th>Min.</th>
<th>1st Qu.</th>
<th>Median</th>
<th>3rd Qu.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.0000e+00</td>
<td>5.9787e-16</td>
<td>9.1551e-16</td>
<td>1.8611e-15</td>
<td>5.5212e-03</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Estimate</th>
<th>StdErr</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>0.08251</td>
<td>0.17</td>
</tr>
<tr>
<td>H3K4me3_signal_bin25</td>
<td>0.00587</td>
<td>0.09</td>
</tr>
<tr>
<td>H3K27me3_signal_bin21</td>
<td>0.01558</td>
<td>0.09</td>
</tr>
</tbody>
</table>

Residual standard error: 0.686 on 21424 degrees of freedom
Multiple \( R^2 \): 0.471, Adjusted \( R^2 \): 0.471
Value of objective function: 0.000

Elapsed time:

<table>
<thead>
<tr>
<th>min</th>
<th>mean</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>11490</td>
<td>11490</td>
<td>11490</td>
</tr>
</tbody>
</table>

From this, it can be seen that the MMM estimators, \( \hat{\beta} \), causes the bandwidth (standard deviation) estimator for FPKM\(_{\lambda} \) and that of the KDE cast over \( \mathbf{X}\beta \), and the values of (4.2.2) – (4.2.4), to be equal. The fitted KDE’s, using Gaussian kernels, with Silverman’s bandwidth estimator, to the transformed “Raw\(_{\text{FPKM}}\)” signal, fitted method of moments estimates, and these results applied to the testing set, along with the respective bandwidth estimators, are illustrated in Figure 4.2. For the training set
\[ r^2 = 0.47082, \text{ and for the testing set } r^2 = 0.47088. \text{ The } r^2 \text{ value as well as the first three moments for the KDE’s of the testing set seem to be stable. This is depicted in the following table:}

\begin{tabular}{|c|c|c|}
\hline
LHS & RHS &  \\
\hline
1 & 0.65208 & 0.65158 \\
2 & 0.85361 & 0.85480 \\
3 & 1.23187 & 1.23051 \\
Silverman BW & 0.10109 & 0.10131 \\
\hline
\end{tabular}

(a) KDE’s for transformed FPKM signal and the method of moments fit thereof, with bandwidths 0.09333 and 0.09333 respectively (training set).

(b) KDE’s for transformed FPKM signal and the method of moments fit thereof, with bandwidths 0.10109 and 0.10131 respectively (testing set).

**Figure 4.2:** Plots for transformed expression data resulting from method of moments.

### 4.5.2 Arc Length Matching

Solving for \( \beta \) using ALR set out in Section 4.3, requires finding solutions to (4.3.1) – (4.3.3). For this, the “alKDEboot” function from the “alR” package is used in solving for \( \beta \). Again, bootstrapping is employed for estimating standard errors, confidence intervals, biases and \( p \)-values for the estimated coefficients. The results now follow.
### Residuals:

<table>
<thead>
<tr>
<th></th>
<th>Min.</th>
<th>1st Qu.</th>
<th>Median</th>
<th>3rd Qu.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-2.17800</td>
<td>-0.27935</td>
<td>-0.11239</td>
<td>0.19860</td>
<td>2.59920</td>
</tr>
</tbody>
</table>

### KDE Arc Lengths:

<table>
<thead>
<tr>
<th></th>
<th>LHS</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-0.12825, 0.43317]</td>
<td>2.50550</td>
<td>2.505496</td>
</tr>
<tr>
<td>[0.43317, 1.75541]</td>
<td>1.41384</td>
<td>1.413842</td>
</tr>
<tr>
<td>BW</td>
<td>0.09333</td>
<td>0.087619</td>
</tr>
</tbody>
</table>

### Value of objective function:

<table>
<thead>
<tr>
<th></th>
<th>Min.</th>
<th>1st Qu.</th>
<th>Median</th>
<th>3rd Qu.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.0000e+00</td>
<td>4.4409e-16</td>
<td>6.6613e-16</td>
<td>1.3506e-15</td>
<td>1.5469e-01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>StdErr</th>
<th>p. value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>0.21554</td>
<td>0.16</td>
<td>&lt;2e-16</td>
</tr>
<tr>
<td>H3K4me3_signal_bin25</td>
<td>0.00641</td>
<td>0.11</td>
<td>&lt;2e-16</td>
</tr>
<tr>
<td>H3K27me3_signal_bin21</td>
<td>-0.00231</td>
<td>0.11</td>
<td>0.19</td>
</tr>
</tbody>
</table>

Residual standard error: 0.489 on 21424 degrees of freedom  
Multiple $R^2$: 0.607, Adjusted $R^2$: 0.607  
Value of objective function: 0.000

Elapsed time:  
<table>
<thead>
<tr>
<th></th>
<th>min</th>
<th>mean</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1377.5</td>
<td>1380.6</td>
<td>1381.9</td>
</tr>
</tbody>
</table>

The ALR estimators, $\hat{\beta}$, do not necessarily cause the bandwidth estimator for the dependent observations, “FPKM_lambda”, and that of the KDE cast over $X\hat{\beta}$ to be equal. The fitted KDE’s, using Gaussian kernels, with Silverman’s bandwidth estimator, to the transformed “Raw_FPKM” signal, fitted ALR estimates, and these results applied to the testing set, along with the respective bandwidth estimators, are illustrated in Figure 4.3. For the training set $r^2 = 0.60716$, and for the testing set $r^2 = 0.63292$. The $r^2$ value as well as the arc length segments for the KDE’s of the testing set seem to be stable. This is depicted in the following table:
Chapter 4. Arc Length Regression

<table>
<thead>
<tr>
<th>LHS</th>
<th>RHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>[−0.1409, 0.40889]</td>
<td>2.36790  2.37996</td>
</tr>
<tr>
<td>[0.40889, 1.75679]</td>
<td>1.44474  1.47711</td>
</tr>
<tr>
<td>BW</td>
<td>0.10109  0.09567</td>
</tr>
</tbody>
</table>

(a) KDE’s for transformed FPKM signal and the arc length regression fit thereof, with bandwidths 0.09333 and 0.08762 respectively (training set).

(b) KDE’s for transformed FPKM signal and the arc length regression fit thereof, with bandwidths 0.10109 and 0.09567 respectively (testing set).

**Figure 4.3:** Plots for transformed expression data resulting from arc length regression.

### 4.6 Model Evaluation

The parameter estimates, along with their properties for the three models fitted to the data, are summarised in Table 4.1. For assessing the respective fit of the models proposed, the $r^2$ values are compared, including a $r^2$ resulting from cross-validation, as well as $D^2$, a measure of divergence between the $\hat{\gamma}$ and $X\hat{\beta}$ vectors. This test is based on the assumption that the two vectors have a common distribution, and similar size, and was proposed by Bhattacharyya (1946) [8]. This assumption seems reasonable, since $\hat{\beta}$ is the set of parameters that provide equal properties for the KDE’s of the two
vectors. Since the sample size in this case is “large”, 100 proportions were compared, and consequently, the \( \chi^2 \) approximation is adequate for approximating the exact distribution derived by Bhattacharyya. The results are shown in Table 4.2.

<table>
<thead>
<tr>
<th></th>
<th>Intercept</th>
<th>H3K4me3_signal_bin25</th>
<th>H3K27me3_signal_bin21</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OLS</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Estimate</td>
<td>0.3630</td>
<td>0.0046</td>
<td>-0.0049</td>
</tr>
<tr>
<td>StdErr</td>
<td>0.0042</td>
<td>0.0000</td>
<td>0.0001</td>
</tr>
<tr>
<td>LCI</td>
<td>0.3548</td>
<td>0.0046</td>
<td>-0.0051</td>
</tr>
<tr>
<td>UCI</td>
<td>0.3712</td>
<td>0.0047</td>
<td>-0.0047</td>
</tr>
<tr>
<td>p.value</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td><strong>MMM</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Estimate</td>
<td>0.0825</td>
<td>0.0059</td>
<td>0.0156</td>
</tr>
<tr>
<td>StdErr</td>
<td>0.1711</td>
<td>0.0935</td>
<td>0.0948</td>
</tr>
<tr>
<td>LCI</td>
<td>0.0749</td>
<td>0.0056</td>
<td>-0.0110</td>
</tr>
<tr>
<td>UCI</td>
<td>0.3114</td>
<td>0.0062</td>
<td>0.0168</td>
</tr>
<tr>
<td>b.value</td>
<td>0.1679</td>
<td>0.0060</td>
<td>0.0052</td>
</tr>
<tr>
<td>Bias</td>
<td>0.0854</td>
<td>0.0001</td>
<td>-0.0103</td>
</tr>
<tr>
<td>p.value</td>
<td>0.7708</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td><strong>ALR</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Estimate</td>
<td>0.2155</td>
<td>0.0064</td>
<td>-0.0023</td>
</tr>
<tr>
<td>StdErr</td>
<td>0.1603</td>
<td>0.1069</td>
<td>0.1105</td>
</tr>
<tr>
<td>LCI</td>
<td>0.0859</td>
<td>0.0056</td>
<td>-0.0041</td>
</tr>
<tr>
<td>UCI</td>
<td>0.2549</td>
<td>0.0066</td>
<td>0.0018</td>
</tr>
<tr>
<td>b.value</td>
<td>0.1912</td>
<td>0.0061</td>
<td>-0.0000</td>
</tr>
<tr>
<td>Bias</td>
<td>-0.0244</td>
<td>-0.0003</td>
<td>0.0023</td>
</tr>
<tr>
<td>p.value</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.1939</td>
</tr>
</tbody>
</table>

Table 4.1: Comparison of bootstrap results for OLS, MMM and ALR methods.

From Table 4.1 it can be seen that the parameter standard errors resulting from OLS are smaller than that of the two newly proposed methods. This is in accordance with the properties of OLS estimates, being those with the smallest variance. The “b.values” is the estimated values with the bias removed, however, these do not correspond to the respective objective functions being minimised. Only the OLS method resulted
Chapter 4. Arc Length Regression

<table>
<thead>
<tr>
<th></th>
<th>OLS</th>
<th>MMM</th>
<th>ALR</th>
</tr>
</thead>
<tbody>
<tr>
<td>r² (Train)</td>
<td>0.5027</td>
<td>0.4708</td>
<td>0.6072</td>
</tr>
<tr>
<td>r² (Test)</td>
<td>0.4966</td>
<td>0.4709</td>
<td>0.6329</td>
</tr>
<tr>
<td>RSS</td>
<td>0.4566</td>
<td>0.6865</td>
<td>0.4890</td>
</tr>
<tr>
<td>D² (Train)</td>
<td>0.3239</td>
<td>0.2566</td>
<td>0.2654</td>
</tr>
<tr>
<td>D² (Test)</td>
<td>0.3294</td>
<td>0.2575</td>
<td>0.2710</td>
</tr>
</tbody>
</table>

Table 4.2: Comparison of performance results for OLS, MMM and ALR methods.

in significant (at the 5% level) \( p \)-values for all parameters; however, these are based on strong underlying model assumptions, whereas the other two methods are non-parametric in nature.

From Table 4.2 it is clear that OLS yielded the smallest residual standard error (indicated by “RSS”), with ALR resulting in the largest \( r^2 \) values. The MMM resulted in the smallest divergences, closely followed by ALR. The \( r^2 \) and \( D^2 \) measures kept steady under cross-validation, and therefore signify stable results. Although the \( p \)-values of the Bhattacharyya test are all very small (\(< 0.0001 \) not given here), the divergence measures on which the test is based, may still be used for comparison purposes.

4.7 Summary

Two new regression techniques were proposed here based on equating properties of KDE’s, based on the set of dependent observations \( y \), and the set of independent observations \( X \beta \). Both these methods didn’t tamper with the linear model configuration, but instead proposed new objective functions. Standard errors, confidence intervals and \( p \)-values were obtained using the bootstrap, and the results obtained compared to that obtained from OLS regression.

It was seen that although OLS regression resulted in the smallest residual standard errors, ALR could predict a larger percentage explained variation as measured by \( r^2 \), but that the MMM, had the smallest divergence between the observed, and predicted value distributions. ALR therefore balances the residual standard error, and sample divergence, while resulting in a larger \( r^2 \).

In the following chapter, the linear model specification will be generalised, to allow for a non-linear regression setting. It will be seen how the arc length may be utilised in
parameter estimation under such assumptions.
Chapter 5

Non-Linear Arc Length Regression

The linear model assumption from Chapter 4 is extended here to the non-linear setting, and the comparison to OLS, naturally replaced by NLS. In this chapter, a NALR technique is proposed, and used to fit the cumulative distribution function (CDF), of a subset of the four-parameter kappa distribution, to time-conductivity curves.

In Section 5.1, the usual model specification for non-linear regression, along with NLS estimation are presented. NALR is then introduced by modification of the NLS objective function.

It is shown that the practical problem in Section 5.2, may be cast in the form of a typical non-linear regression setting, solved in Section 5.3, using a subset of the four-parameter kappa CDF. This is obtained by enforcing a particular relationship between its two shape parameters. Finally, the NLS and NALR techniques are demonstrated in Section 5.4, using data resulting from actual experiments.

5.1 Introduction

Consider \( n \) pairs of observations, \((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\) for which

\[
y_i = F(x_i, \beta)
\]

describes the error-free relationship between a set of dependent and independent observations \( y \), and \( x \), respectively. Here, \( F(\cdot) \) is a non-linear function in the \( p \) parameters \( \beta \), say.
In NLS, the estimated parameters $\hat{\beta}$, is the set of parameter values that minimises the objective function

$$SSE = \sum (y - F(x, \hat{\beta}))^2.$$  

Define

$$S(a, b, \beta) = \int_a^b \sqrt{1 + (f(x, \beta))^2} \, dx$$

as the arc length of the function $F(x, \beta)$ on the interval $[a, b]$. It is assumed that $f(x, \beta) = F'(x, \beta)$ exists and is well defined. Also define

$$f(a, b) = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (5.1.1)$$

to be the sample arc length statistic. Here, $(x(i), y(i))$ denotes the observed pairs, ordered in $x$, i.e. $x(1) \leq \ldots \leq x(n)$. This is done in order to honour the relationship between $x$ and $y$. The estimated parameters $\hat{\beta}$ is the set of parameters that minimises the objective function

$$\left( S(a, b, \hat{\beta}) - f(a, b) \right)^2.$$ 

The interval $[a, b]$ may be thought of as tuning parameters, and may even correspond to vectors, so that the arc lengths over a collection of intervals are matched. The objective function for a collection of $n$ arc length segments then becomes

$$\sqrt{\sum_{i=1}^{n} \left( S(a_i, b_i, \hat{\beta}) - f(a_i, b_i) \right)^2}.$$  

The optimisation process is therefore similar to that of NLS, with a change in only the objective function.

### 5.2 Problem Description

Currently the reference method for the measurement of the oxidative stability of biodiesel is the EUROPEAN STANDARD EN 14112 (2003) [27]. It prescribes the Rancimat method by Hadorn (1974) [39] and Lübli and Bruttel (1986) [51] for measuring the
induction time of the biodiesel. In this method autoxidation is accelerated by passing a constant flow of air through the biodiesel sample and controlling the temperature at an elevated level, i.e. 110 °C. The oxidation process is driven by radical reactions that involve the unsaturated fatty acids. During an initial induction phase virtually no secondary products are formed. This is abruptly followed by an oxidation phase characterised by a rapid increase in peroxide value and the formation of volatile products. The Rancimat method relies on the fact that the greater part of these volatile matter consists of formic acid. This is trapped by passing the air through distilled water and its accumulation is recorded conductometrically. The length of the induction period (IP) is taken as a measure of oxidative stability. The EUROPEAN STANDARD EN 14214 (2008) [28] describes two methods for the evaluation of the IP. Furthermore there is a tacit assumption that the two methods yield comparable if not identical results.

Both these methods utilise the conductivity vs. time curve, say $F(x)$. Focke, van der Westhuizen and Oosthuizen (2016) [32] showed that this process can be automated by curve fitting using a range of sigmoidal functions, however, that analysis demonstrated that the IP values resulting from the two methods differ. The question now is to establish conditions under which the two methods yield equivalent results.

Let $X$ be a unimodal random variable with CDF and PDF respectively $F(X)$ and $f(x)$. By definition

$$f(x) = F'(x).$$

The mode of $X$ is attained at

$$f'(x) = F''(x) = 0,$$

say at $c$, i.e. the inflection point of $F(x)$ or global maximum of $f(x)$.

For the first method, $F'(x) = f(x)$ is the gradient of $F(x)$ at any point $x$. The formula for the gradient of the straight line at the point $c$ is given by $f(c)$. $c$ corresponds to the mode of $X$, and may be found as the inflection point $(c,0)$, say, of $F(x)$, i.e. where

$$F''(c) = 0. \quad (5.2.1)$$

The formula for the straight line, $T(x)$, through the point $(c,F(c))$, with gradient $f(c)$
is therefore given by

\[ F(c) = f(c)c + c_0 \]
\[ \therefore c_0 = F(c) - f(c)c \]
\[ \therefore T(x) = f(c) (x - c) + F(c). \] (5.2.2)

The IP value is then the point \((a, 0)\) where this intersects the \(x\)-axis (see EUROPEAN STANDARD EN 14214 (2008) [28]), and is given by

\[ T(a) = 0 \]
\[ \therefore 0 = f(c) (a - c) + F(c) \]
\[ \therefore a = c - \frac{F(c)}{f(c)}. \] (5.2.3)

For the second method, suppose that the IP value is \((b, 0)\) which is the smallest solution of

\[ F'''(b) = 0, \] (5.2.4)

(see EUROPEAN STANDARD EN 14214 (2008) [28]).

In addressing the research question, a CDF, \(F(x)\), is therefore sought that satisfies

\[ a = c - \frac{F(c)}{f(c)} = b, \]

or if \(a = b = d\), say, then from (5.2.1), (5.2.3), and (5.2.4):

\[ F''(c) = F'''(d) = F''' \left( c - \frac{F(c)}{f(c)} \right). \]

As an example, consider the standard normal distribution, which has mode \(c = 0\). For method 1, the line through the point \((c, F(c)) = (0, 0.5)\), with gradient \(f(c) = f(0) = \frac{1}{\sqrt{2\pi}}\) is given by

\[ T(x) = \frac{x}{\sqrt{2\pi}} + 0.5, \]

with corresponding IP value using (5.2.3)

\[ a = -\sqrt{\frac{\pi}{2}} = -1.2533141373. \]
For the second method, the calculations reduce to

\[ F'''(x) = f''(x) \]
\[ f'(x) = -xf(x) \]
\[ \therefore f''(x) = -f(x) - xf'(x) \]
\[ = -f(x) + x^2 f(x) \]
\[ = f(x)(x^2 - 1) \]
\[ \therefore F'''(x) = 0 \]
\[ \Leftrightarrow 0 = f(x)(x^2 - 1) \]
\[ \therefore x = \pm 1. \]

\( b \) is now the smallest of these two roots, namely \( b = -1 \).

Thus, \( a \) corresponds to the point \((-1.2533141373, 0)\) which is less than \( b \) corresponding to the point \((-1, 0)\). Clearly the standard normal distribution isn’t a solution to this problem. This solution is depicted in Figure 5.1.
Figure 5.1: Illustration of the two methods for the evaluation of the IP.

In summary then, taking into account (5.2.1), (5.2.3), and (5.2.4), a suitable probability distribution will exhibit the following properties:

- \( f'' \left( c - \frac{F(c)}{f(c)} \right) = 0 \), where

- \( F''(c) = 0 \), i.e. \( c \) is the mode of the distribution.
5.3 The Four-Parameter Kappa Distribution

The four-parameter kappa distribution was introduced by Hosking (1994) [45] as a generalisation of the three-parameter version, introduced by Mielke Jr (1973) [60], by making use of a suitable re-parameterisation. In order to simplify notation, let
\[ g(x) = 1 - k \frac{x - \mu}{\sigma} \]
\[ \therefore g'(x) = - \frac{k}{\sigma}, \]
then the quantile function (QF), CDF, and PDF are respectively given by
\[
Q(p) = \mu + \frac{\sigma}{k} \left( 1 - \left( \frac{1 - p^h}{h} \right)^k \right), \quad 0 \leq p \leq 1,
\]
\[
F(x) = \left( 1 - h \left( 1 - k \frac{x - \mu}{\sigma} \right)^{1/k} \right)^{1/h} = \left( 1 - h (g(x))^{1/k} \right)^{1/h}
\]
\[
f(x) = - \frac{1}{k} \frac{F(x)}{\left( 1 - h (g(x))^{1/k} \right)} (g(x))^{1/k-1} g'(x) = \frac{1}{\sigma} \frac{F(x)}{\left( 1 - h (g(x))^{1/k} \right)} (g(x))^{1/k-1} = \frac{1}{\sigma} \left( 1 - h (g(x))^{1/k} \right)^{1/h-1} (g(x))^{1/k-1} = \frac{1}{\sigma} (F(x))^{1-h} (g(x))^{1/k-1}.
\]
Here \( \mu \in \mathbb{R} \) is a location parameter, \( \sigma > 0 \) is a scale parameter, and \( h, k \in \mathbb{R} \) are shape parameters. The notation used here is ment to include the limiting cases for \( h = 0 \) and
\[ k = 0. \] The first derivative is then

\[
f'(x) = (1 - h) (F(x))^{-1} (f(x))^2 - \frac{1 - k}{\sigma} f(x) (g(x))^{-1} \]

\[
= f(x) \left[ (1 - h) (F(x))^{-1} f(x) - \frac{1 - k}{\sigma} (g(x))^{-1} \right] \]

\[
= f(x) \left[ (1 - h) (F(x))^{-1} \frac{1}{\sigma} (F(x))^{1-h} (g(x))^{1/k} - \frac{1 - k}{\sigma} (g(x))^{-1} \right] \]

\[
= \frac{1}{\sigma} f(x) (g(x))^{-1} \left[ (1 - h) (F(x))^{-h} (g(x))^{1/k} - (1 - k) \right] \]

\[
= \frac{1}{\sigma} f(x) (g(x))^{-1} \left[ (1 - h) \left( 1 - h (g(x))^{1/k} \right)^{-1} (g(x))^{1/k} - (1 - k) \right] \]

\[
= \frac{1}{\sigma} f(x) (g(x))^{-1} \left[ (1 - h) \left( (g(x))^{-1/k} - h \right)^{-1} - (1 - k) \right]. \]

The mode is therefore given by

\[
f'(x) = 0 \implies \left( \frac{1 - k}{1 - hk} \right)^k = g(x) \]

\[
\therefore x = \mu + \frac{\sigma}{k} \left( 1 - \left( \frac{1 - k}{1 - hk} \right)^k \right) = c, \text{ say.} \quad (5.3.1) \]
The second derivative is given by

\[
f''(x) = (1 - h) \left[ - (F(x)^{-2} (f(x))^3 + 2 (F(x))^{-1} f(x) f'(x) ight. \\
- \frac{1 - k}{\sigma} \left[ f'(x) (g(x))^{-1} - f(x) (g(x))^{-2} g'(x) \right] \\
= (1 - h) (F(x))^{-1} f(x) \left[ 2 f'(x) - (F(x))^{-1} (f(x))^2 \right] \\
- \frac{1 - k}{\sigma} (g(x))^{-1} \left[ f'(x) + \frac{k}{\sigma} f(x) (g(x))^{-1} \right] \\
= (1 - h) (F(x))^{-1} \frac{1}{\sigma} (F(x))^{1-h} (g(x))^{1/k-1} \\
\times \left[ 2 f'(x) - (F(x))^{-1} \left( \frac{1}{\sigma} (F(x))^{1-h} (g(x))^{1/k-1} \right)^2 \right] \\
- \frac{1 - k}{\sigma} (g(x))^{-1} \left[ f'(x) + \frac{k}{\sigma} (F(x))^{1-h} (g(x))^{1/k-1} (g(x))^{-1} \right] \\
= \frac{1 - h}{\sigma} (F(x))^{-h} (g(x))^{1/k-1} \\
\times \left[ 2 f'(x) - \frac{1}{\sigma^2} (F(x))^{1-2h} (g(x))^{2/k-2} \right] \\
- \frac{1 - k}{\sigma} (g(x))^{-1} \left[ f'(x) + \frac{k}{\sigma^2} (F(x))^{1-h} (g(x))^{1/k-2} \right] \\n= (g(x))^{-1} \left\{ \frac{1 - h}{\sigma} (F(x))^{-h} (g(x))^{1/k} \\
\left[ 2 f'(x) - \frac{1}{\sigma^2} (F(x))^{1-2h} (g(x))^{2/k-2} \right] \\
- \frac{1 - k}{\sigma} \left[ f'(x) + \frac{k}{\sigma^2} (F(x))^{1-h} (g(x))^{1/k-2} \right] \right\}.
\]

Since this final expression cannot be simplified further, and since its roots cannot be found analytically, numerical solutions will be employed.

If the value for \( c \) given in (5.3.1), is substituted into (5.2.3), it follows that

\[
g(c) = \left( \frac{1 - k}{1 - hk} \right)^{1/k} \\
F(c) = \left( \frac{1 - h}{1 - hk} \right)^{1/h} \\
f(c) = \frac{1}{\sigma} \left( \frac{1 - h}{1 - hk} \right)^{1/(h-1)} \left( \frac{1 - k}{1 - hk} \right)^{1-k}
\]
\[
\therefore \frac{F(c)}{f(c)} = \sigma \left( \frac{1 - h}{1 - hk} \right)^{1/h} \left( \frac{1 - h}{1 - hk} \right)^{1-1/h} \left( \frac{1 - k}{1 - hk} \right)^{k-1} \\
F(c) = \frac{1 - h}{1 - k} \left( \frac{1 - k}{1 - hk} \right)^k \\
\therefore c - \frac{F(c)}{f(c)} = \mu + \frac{\sigma}{k} - \sigma \left( \frac{1 - k}{1 - hk} \right)^k \left( \frac{1}{k} + \frac{1 - h}{1 - k} \right) \\
= \mu + \frac{\sigma}{k} - \sigma \left( \frac{1 - k}{1 - hk} \right)^k \frac{1}{k} \left( \frac{1 - k}{1 - hk} \right)^{-1} \\
= \mu + \frac{\sigma}{k} \left[ 1 - \left( \frac{1 - k}{1 - hk} \right)^{k-1} \right].
\]

(5.3.2)

All of the derivatives of \(F(\cdot)\) may be written in terms of \(g(\cdot)\) (using (5.3.1)), and since the substitution of (5.3.2) into \(g(\cdot)\) will yield a result that is independent of the choice of \(\mu\) or \(\sigma\), \((h, k)\) combinations can be calculated numerically, such that

\[
f'' \left( c - \frac{F(c)}{f(c)} \right) = 0.
\]

In order for the four-parameter kappa distribution to be a suitable solution to the problem, the “allR” package in “R” by Loots (2017) \[54\] is used for obtaining \((h, k)\) combinations that yield suitable shape parameters. The Nelder-Mead simplex method, by Nelder and Mead (1965) \[67\], is used in order to minimise

\[
\left| f'' \left( c - \frac{F(c)}{f(c)} \right) \right|.
\]

(5.3.3)

Since a distribution is sought which has a global maximum of the PDF at \(c\), it is noted from Hosking (1994) \[45\] that \(f(x)\) has a single maximum when

\[
h \in (-\infty, 0) \text{ and } k \in \left( \frac{1}{h}, 1 \right), \text{ or } \]

\[
h \in [0, 1) \text{ and } k \in (-\infty, 1).
\]

Thus, \(h \in (-\infty, 1)\), and \(k \in (-\infty, 1)\). Furthermore, since the variable of interest is time, the distribution should have positive support, and should be bounded below. Following from the optimisation procedure, the following configurations are possible:
Table 5.1: \((h, k)\) combinations for the four-parameter kappa distribution.

<table>
<thead>
<tr>
<th>Bound</th>
<th>Support</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(x \in [\mu + \sigma \frac{1-h}{k}, \mu + \frac{\sigma}{k}])</td>
<td>(h &gt; 0, k &gt; 0)</td>
</tr>
<tr>
<td>2</td>
<td>(x \in [\mu + \sigma \ln h, \infty))</td>
<td>(h &gt; 0, k = 0)</td>
</tr>
<tr>
<td>3</td>
<td>(x \in [\mu + \sigma \frac{1-h}{k}, \infty))</td>
<td>(h &gt; 0, k &lt; 0)</td>
</tr>
<tr>
<td>4</td>
<td>(x \in (-\infty, \mu + \frac{\sigma}{k}])</td>
<td>(h \leq 0, k &gt; 0)</td>
</tr>
<tr>
<td>5</td>
<td>(x \in (-\infty, \infty))</td>
<td>(h \leq 0, k = 0)</td>
</tr>
<tr>
<td>6</td>
<td>(x \in [\mu + \frac{\sigma}{k}, \infty))</td>
<td>(h \leq 0, k &lt; 0)</td>
</tr>
</tbody>
</table>

Figure 5.2: \((h, k)\) combinations for the four-parameter kappa distribution that minimise (5.3.3).
From Figure 5.2, this is a decreasing function of $k$ for $h < -0.05$, and increasing when $h \geq -0.05$. There is a discontinuity at $h = 0$, which corresponds to a generalised extreme value distribution (GEV). $h < 0$ and $k < 0$ yield a Burr type III distribution. A special case of this is when $h = -1$ which renders a generalised logistic distribution. Bound 6 from Table 5.1 corresponds to this configuration, i.e. distributions bounded below, but not from above. $h > 0$ and $k > 0$ refer to distributions between the generalised extreme value, and generalised Pareto distributions. For this, Bound 1 from Table 5.1 applies, which is bounded above and below. For more information on these special cases see Hosking (1994) [45].

The relationship between $h$ and $k$ clearly indicates that none of the other special cases such as the exponential ($h = 1$ and $k = 0$), Gumbel ($h = 0$ and $k = 0$), logistic ($h = -1$ and $k = 0$), uniform ($h = 1$ and $k = 1$), or reverse exponential ($h = 0$ and $k = 1$) distributions are candidates for being a solution to the problem at hand. As noted above, from Table 5.1, Bound 2 – 4 seems unlikely to be observed, however, Bound 5, with infinite support, never occurs, since $k = 0$ is observed when $h > 0$.

5.4 Results

Two samples of sizes $n_1 = 5770$, and $n_2 = 4080$ are considered respectively. This correspond to two experiments at 0.33%, 0%, and 0.67% concentrations of antioxidants Orox PK, Naugard P and Anox 20 respectively. The sigmoidal (or time vs. conductivity) curves are shown in Figure 5.3.
In order to fit a CDF to this data, the data should be rescaled so that $y \in [0, 1]$, i.e. all $y$ values have to be divided by

$$\tau(\beta) = F(\max x, \beta).$$

This truncation may be considered as a dynamic process of scaling, since it depends on the parameter vector $\beta$, which is to be estimated, and has no effect on the mode of the distribution, nor on the conditions for the method at hand (as derived in Section 5.3).

For fitting the CDF of the (right-truncated) four-parameter kappa distribution to data, two boundary specifications are considered separately, for $h \leq 0$ and $h > 0$, corresponding to Bound 1 and 6 from Table 5.1. This is done, in order to eliminate the estimation of $\mu$. Since the data is bounded from below, say at $x_0$, $\mu$ is chosen such that:

$h < 0$

$$x_0 = \mu + \frac{\sigma}{k}$$

$$\therefore \mu = x_0 - \frac{\sigma}{k},$$

$h > 0$

$$x_0 = \mu + \sigma \frac{1 - h^{-k}}{k}$$

$$\therefore \mu = x_0 - \sigma \frac{1 - h^{-k}}{k}$$
Chapter 5. Non-Linear Arc Length Regression

The case where \( h = 0 \) and \( k < 0 \) will be considered if it seems that the estimate for \( h \) converges to zero.

### 5.4.1 Implementation Infrastructure

The “DEoptimR” package in “R” by Conceicao and Maechler (2015) [15] provides support for differential evolution with non-linear constraints, and was used to obtain NLS and NALR estimates, since it provides global solutions to the respective objective functions. These solutions were consequently used as starting values for the “constrOptim.nl” function from the “alabama” package by Varadhan (2015) [82]. This ensures that bootstrap solutions converge to a neighbourhood containing the global solution, and also increases the calculations involved.

### 5.4.2 Sample1

<table>
<thead>
<tr>
<th></th>
<th>mu</th>
<th>sigma</th>
<th>h</th>
<th>k</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLS</td>
<td>12.4186</td>
<td>4.7450</td>
<td>-0.3632</td>
<td>-0.3821</td>
</tr>
<tr>
<td></td>
<td>0.0295</td>
<td>0.0225</td>
<td>0.0053</td>
<td>0.0009</td>
</tr>
<tr>
<td></td>
<td>12.3776</td>
<td>4.7226</td>
<td>-0.3729</td>
<td>-0.3849</td>
</tr>
<tr>
<td></td>
<td>12.4855</td>
<td>4.8051</td>
<td>-0.3531</td>
<td>-0.3816</td>
</tr>
<tr>
<td></td>
<td>12.4305</td>
<td>4.7632</td>
<td>-0.3627</td>
<td>-0.3833</td>
</tr>
<tr>
<td></td>
<td>0.0119</td>
<td>0.0182</td>
<td>0.0005</td>
<td>-0.0012</td>
</tr>
<tr>
<td></td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>NALR</td>
<td>10.8447</td>
<td>4.1225</td>
<td>-0.7172</td>
<td>-0.3801</td>
</tr>
<tr>
<td></td>
<td>2.3827</td>
<td>0.5250</td>
<td>0.2182</td>
<td>0.0268</td>
</tr>
<tr>
<td></td>
<td>6.7586</td>
<td>2.6503</td>
<td>-1.5606</td>
<td>-0.3923</td>
</tr>
<tr>
<td></td>
<td>17.8021</td>
<td>4.5204</td>
<td>-0.5622</td>
<td>-0.2712</td>
</tr>
<tr>
<td></td>
<td>10.5807</td>
<td>3.6665</td>
<td>-0.8978</td>
<td>-0.3556</td>
</tr>
<tr>
<td></td>
<td>-0.2640</td>
<td>-0.4560</td>
<td>-0.1806</td>
<td>0.0245</td>
</tr>
<tr>
<td></td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0330</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLS</td>
<td>0.0119</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>NALR</td>
<td>-0.2640</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>0.0330</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLS</td>
<td>-0.2640</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>0.0330</td>
<td>0.0000</td>
</tr>
<tr>
<td>NALR</td>
<td>-0.2640</td>
<td>0.0000</td>
</tr>
<tr>
<td></td>
<td>0.0330</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

**Table 5.2:** Comparison of bootstrap results for NLS and NALR methods for sample1.
Chapter 5. Non-Linear Arc Length Regression

<table>
<thead>
<tr>
<th></th>
<th>NLS</th>
<th>NALR</th>
</tr>
</thead>
<tbody>
<tr>
<td>r² (Train)</td>
<td>0.9977</td>
<td>0.9629</td>
</tr>
<tr>
<td>r² (Test)</td>
<td>0.9980</td>
<td>0.9653</td>
</tr>
<tr>
<td>RSS</td>
<td>1.1054</td>
<td>5.8181</td>
</tr>
<tr>
<td>rMSE</td>
<td>0.0187</td>
<td>0.0984</td>
</tr>
<tr>
<td>Zero</td>
<td>1.0000E-05</td>
<td>-6.9690E-06</td>
</tr>
<tr>
<td>τ</td>
<td>0.7399</td>
<td>0.8142</td>
</tr>
<tr>
<td>IP (7.55)</td>
<td>6.4590</td>
<td>4.4802</td>
</tr>
</tbody>
</table>

Table 5.3: Comparison of performance results for NLS and AL methods for sample1.

Table 5.2 suggests that a four-parameter kappa distribution with parameters \( \mu = 12.4185870463, \sigma = 4.744973998, h = -0.3632040805, \) and \( k = -0.3820864628, \) implying a Burr type III distribution, fits the data the best, as obtained by NLS estimation. Although the \( p \)-values obtained from NALR, (with the arc length computed using the endpoints of the interval corresponding to the 0.7\textsuperscript{th} and 0.982\textsuperscript{nd} empirical quantiles of the data) are all significant at the \( \alpha = 0.05 \) level, the coefficient standard errors, biases, and confidence intervals for NLS estimation are smaller. Table 5.3 supports this fact by showing that the corresponding residual standard error (RSS) and rMSE values for NLS estimation are smaller than that of NALR. The \( r^2 \) values hold steady for both methods in the validation sample, and therefore provides evidence of a reasonably good fit. The “Zero” row indicates that the constrained optimisation is satisfied for both methods. The model suggested by the NLS method truncates the process at \( \tau = 0.7399284597, \) and NALR at \( \tau = 0.8141514862, \) i.e. further along. The IP value corresponding to the NLS method of 6.4590059957, is closer to the automated machine produced value of 7.55, than the NALR value of 4.4802030506. As a check of the validity of the results, the mode of the NLS model is attained at 10.3653615566, and that of the NALR model at 8.5010505495, which are both larger than the corresponding IP values, but smaller than the maximum value of 19.23243667.
5.4.3 Sample 2

Table 5.4 suggests that a four-parameter kappa distribution with parameters $\mu = 13.3572858936$, $\sigma = 5.0933422015$, $h = -0.3529822585$, and $k = -0.3813156536$ fits the data the best, as obtained by NALR. Here the endpoints of the interval over which the arc length was computed, corresponded to the 0.3$^{\text{th}}$ and 0.63$^{\text{th}}$ empirical quantiles of the data. Although the $p$-values obtained from both NLS and NALR are all significant, the coefficient standard errors, biases, and confidence intervals for NALR are smaller. Table 5.5 shows

\begin{figure}[h]
\centering
\begin{subfigure}[b]{0.45\textwidth}
\centering
\includegraphics[width=\textwidth]{sample1_training.png}
\caption{NLS: sample1 (training)}
\end{subfigure}\hfill
\begin{subfigure}[b]{0.45\textwidth}
\centering
\includegraphics[width=\textwidth]{sample1_testing.png}
\caption{NLS: sample1 (testing)}
\end{subfigure}
\begin{subfigure}[b]{0.45\textwidth}
\centering
\includegraphics[width=\textwidth]{sample1_training_nalr.png}
\caption{NALR: sample1 (training)}
\end{subfigure}\hfill
\begin{subfigure}[b]{0.45\textwidth}
\centering
\includegraphics[width=\textwidth]{sample1_testing_nalr.png}
\caption{NALR: sample1 (testing)}
\end{subfigure}
\caption{Fitted sigmoidal curves to sample1.}
\end{figure}
Table 5.4: Comparison of bootstrap results for NLS and NALR methods for sample2.

<table>
<thead>
<tr>
<th></th>
<th>mu</th>
<th>sigma</th>
<th>h</th>
<th>k</th>
</tr>
</thead>
<tbody>
<tr>
<td>NLS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Estimate</td>
<td>12.4186</td>
<td>4.7450</td>
<td>-0.3632</td>
<td>-0.3821</td>
</tr>
<tr>
<td>StdErr</td>
<td>0.0295</td>
<td>0.0225</td>
<td>0.0053</td>
<td>0.0009</td>
</tr>
<tr>
<td>LCI</td>
<td>12.3776</td>
<td>4.7226</td>
<td>-0.3729</td>
<td>-0.3849</td>
</tr>
<tr>
<td>UCI</td>
<td>12.4855</td>
<td>4.8051</td>
<td>-0.3531</td>
<td>-0.3816</td>
</tr>
<tr>
<td>b.value</td>
<td>12.4305</td>
<td>4.7632</td>
<td>-0.3627</td>
<td>-0.3833</td>
</tr>
<tr>
<td>Bias</td>
<td>0.0119</td>
<td>0.0182</td>
<td>0.0005</td>
<td>-0.0012</td>
</tr>
<tr>
<td>p.value</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NALR</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Estimate</td>
<td>13.3573</td>
<td>5.0933</td>
<td>-0.3530</td>
<td>-0.3813</td>
</tr>
<tr>
<td>StdErr</td>
<td>0.0227</td>
<td>0.0000</td>
<td>0.0022</td>
<td>0.0006</td>
</tr>
<tr>
<td>LCI</td>
<td>13.3197</td>
<td>5.1258</td>
<td>-0.3581</td>
<td>-0.3852</td>
</tr>
<tr>
<td>UCI</td>
<td>13.4390</td>
<td>5.1259</td>
<td>-0.3533</td>
<td>-0.3816</td>
</tr>
<tr>
<td>b.value</td>
<td>13.4175</td>
<td>5.1259</td>
<td>-0.3554</td>
<td>-0.3821</td>
</tr>
<tr>
<td>Bias</td>
<td>0.0602</td>
<td>0.0325</td>
<td>-0.0024</td>
<td>-0.0008</td>
</tr>
<tr>
<td>p.value</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

that although the corresponding RSS and rMSE values for NLS estimation are smaller than that of NALR, the $r^2$ values for NALR are more stable in a cross-validation setting than that of NLS estimation, and therefore provides evidence of over-fitting on the part of the latter. The “Zero” row indicates that the constrained optimisation is satisfied for both methods. The model suggested by the NLS method truncates the process at $\tau = 0.499876692$, and NALR at $\tau = 0.4394090106$, i.e. earlier in the process. The IP value corresponding to the NALR technique of 7.0017681635, is closer to the automated machine produced value of 7.20, than the NLS value of 6.4590059957. As a check of the validity of the results, the mode of the NLS model is attained at 10.3653615566, and that of the NALR model at 11.1759022681, which are both larger than the corresponding IP values, but smaller than the maximum value of 13.59872167.
### Table 5.5: Comparison of performance results for NLS and AL methods for sample2.

<table>
<thead>
<tr>
<th>Metric</th>
<th>NLS</th>
<th>NALR</th>
</tr>
</thead>
<tbody>
<tr>
<td>r2 (Train)</td>
<td>0.9977</td>
<td>0.9953</td>
</tr>
<tr>
<td>r2 (Test)</td>
<td>0.9942</td>
<td>0.9961</td>
</tr>
<tr>
<td>RSS</td>
<td>1.1054</td>
<td>1.7258</td>
</tr>
<tr>
<td>rMSE</td>
<td>0.0187</td>
<td>0.0345</td>
</tr>
<tr>
<td>Zero</td>
<td>1.0000E-05</td>
<td>1.4158E-06</td>
</tr>
<tr>
<td>τ</td>
<td>0.4999</td>
<td>0.4394</td>
</tr>
<tr>
<td>IP (7.20)</td>
<td>6.4590</td>
<td>7.0018</td>
</tr>
</tbody>
</table>

Figure 5.5: Fitted sigmoidal curves to sample2.
5.5 Summary

As an alternative to NLS, NALR was introduced in this chapter. The (truncated) four-parameter kappa CDF was considered as the non-linear function to be fitted, under a set of constraints. These constraints specifically resulted from biodiesel experiments where the induction period (IP) was of interest, obtained using the Rancimat method. From the results obtained, it was clear that circumstances exist under which NLS regression could be improved upon, and where NALR resulted in a plausible alternative. Both the NLS and NALR techniques recommended the Burr type III family of distributions as a solution to the problem posed here, although a much larger family was allowed for initially. The NALR technique provides for fitting models only on a subset of the data in a natural fashion, and is therefore ideal in situations where the data includes a natural drift, as was the case here.

In the following chapter, the mathematical premises for the estimation and regression techniques proposed in this thesis, are provided.
Chapter 6

A Theoretical Framework for Arc Length Based Statistics

In Chapters 3 – 5 arc length based measures have been proposed for parameter estimation (Section 3.3), GoF tests (Section 3.4), coefficient estimation in linear (Section 4.3) and non-linear (Chapter 5) regression. All these techniques make an implicit assumption of uniqueness of the arc length, in that it represents a characterisation of a random process.

This characterisation is presented in Section 6.1, and applied to the uniform distribution in Section 6.2, and exponential distribution in Section 6.3.

6.1 Introduction

Proschan and Pyke (1967) [70] proposed a test for monotone failure rate, based on spacings. This was further investigated by Bickel and Doksum (1969) [10] and Bickel (1969) [9]. It makes therefore sense to investigate the arc lengths of hazard functions, as the proposed sample arc length statistics are based on spacings (see Chapter 1).

Suppose that $T$ is a continuous univariate random variable, with PDF $g(t)$, and CDF $G(t)$, with support $D \subseteq \mathbb{R}$. The arc length of $g(t)$ on the interval $[a, b] \subseteq D$ is defined as

$$S^{[a, b]}_g = \int_a^b \sqrt{1 + (g'(t))^2}dt$$

where $g'(t)$ denotes the first derivative of $g(t)$ with respect to $t$. This corresponds to Definition 2.1.1, but has been reproduced here for completeness.
Proposition 6.1.1 (Arc Length Characterisation). Suppose that \( T \) is a continuous unimodal, univariate random variable with support on the interval \( D = [t_0, t_1] \subset \mathbb{R} \), with PDF and CDF \( g(t) \) and \( G(t) \) respectively, and that \( g'(t) \) exists. The arc length transformation
\[
h(t) = \sqrt{1 + (g'(t))^2}
\] (6.1.1)
defines a pseudo hazard function which characterises a class of distributions, having \( f(t) \) and \( F(t) \) respectively in the following way
\[
F(t) = \frac{1 - e^{-H(t)}}{1 - e^{-H(t_1)}}, \quad \text{and} \quad f(t) = \frac{h(t)e^{-H(t)}}{1 - e^{-H(t_1)}}.
\]

Proof. The result follows from the general solution to the first-order linear ordinary inhomogeneous differential equation with variable coefficients. Adopting the cumulative hazard function
\[
H(t) = \int_{t_0}^{t} h(x)dx
\]
notation, with the understanding that \( h(t) \) (see (6.1.1)) is strictly speaking not a hazard function, but will be referred to as a pseudo hazard function, since
\[
\lim_{t \to \infty} H(t) < \infty.
\]
It follows that

\[ h(t) = \frac{Af(t)}{1 - AF(t)} \]

\[ = \frac{AF'(t)}{1 - AF(t)} \]

\[ \therefore AF'(t) = h(t)(1 - AF(t)) \]

\[ \therefore AF'(t) + h(t)AF(t) = h(t) \]

\[ \therefore F'(t) + h(t)F(t) = \frac{1}{A} h(t) \]

\[ \therefore F'(t)e^{\int h(x)dx} + h(t)F(t)e^{\int h(x)dx} = \frac{1}{A} h(t)e^{\int h(x)dx} \]

\[ \therefore F'(t)e^{H(t)} + h(t)F(t)e^{H(t)} = \frac{1}{A} h(t)e^{H(t)} \]

\[ \therefore \frac{d}{dt} \left( F(t)e^{H(t)} \right) = \frac{1}{A} h(t)e^{H(t)} \]

\[ \therefore F(t) = \frac{1}{A} e^{-H(t)} \left[ \int_{t_0}^{t} h(x)e^{H(x)}dx + C \right] \]

\[ = \frac{1}{A} e^{-H(t)} \left[ e^{H(t)} - 1 + C \right], \] (6.1.2)

where \( A \) and \( C \) are constants. This is subject to the constraints that \( F(t_0) = 0 \), and \( F(t_1) = 1 \). It follows that

\[ 0 = \frac{1}{A} e^{-H(t_0)} \left[ e^{H(t_0)} - 1 + C \right] \]

\[ \therefore C = 0 \]

\[ 1 = \frac{1}{A} e^{-H(t_1)} \left[ e^{H(t_1)} - 1 + C \right] \]

\[ A = e^{-H(t_1)} \left[ e^{H(t_1)} - 1 \right] \]

\[ = 1 - e^{-H(t_1)}. \]

Therefore, from (6.1.2)

\[ F(t) = \frac{1}{1 - e^{-H(t_1)}} e^{-H(t)} \left( e^{H(t)} - 1 \right) \]

\[ = \frac{1 - e^{-H(t)}}{1 - e^{-H(t_1)}} \]

\[ \therefore f(t) = \frac{h(t)e^{-H(t)}}{1 - e^{-H(t_1)}}. \]
Corollary 6.1.2. Suppose that $T$ is a continuous univariate random variable with semi-infinite support $D = [t_0, \infty) \subseteq \mathbb{R}$, with PDF $g(t)$, and CDF $G(t)$, and that $g'(t)$ exists. The arc length transformation

$$h(t) = \sqrt{1 + (g'(t))^2}$$

defines a hazard function which characterises a class of distributions, having $f(t)$, and $F(t)$ as PDF and CDF respectively, in the following way

$$F(t) = 1 - e^{-H(t)}.$$

Proof. From standard definitions of hazard functions

$$h(t) = \frac{f(t)}{1 - F(t)} = - \frac{d}{dt} \ln(1 - F(t))$$

.$$F(t) = 1 - e^{- \int_0^t h(x) dx}$$

$$= 1 - e^{-H(t)}.$$  

note that

$$\lim_{t \to \infty} H(t) \to \infty$$

so that the hazard function $h(t)$ is well defined.

Proposition 6.1.1 and Corollary 6.1.2 assert that $T$ may be uniquely characterised by the full set of all possible arc length segments of $g(t)$ on its domain. The uniqueness sprouts from the association with $F(t)$, the CDF of a class of transformed variates, related to $T$ through an arc length transformation, and by recognising

$$S_g^{[t_0,t]} = H(t).$$

The uniform distribution is used in illustrating Proposition 6.1.1 as it is bounded, whereas the exponential distribution will serve as an example of a distribution with only a lower bound, in illustrating Corollary 6.1.2.
6.2 Uniform Distribution

Let \( T \sim \text{Uniform}(\alpha, \beta), \alpha \leq t \leq \beta \), i.e.

\[
g(t) = \frac{1}{\beta - \alpha}, \quad \alpha \leq t \leq \beta
\]

so that the arc length on any arbitrary interval \([a, b] \in [\alpha, \beta]\) is given by

\[
S_g^{[a,b]} = \int_a^b \sqrt{1 + (g'(x))^2} \, dx
\]

\[
= \int_a^b dx
\]

\[
= b - a,
\]

since the PDF of the continuous uniform distribution is just a horizontal line. Thus from Proposition 6.1.1, for \( \alpha \leq t \leq \beta \),

\[
F(t) = \frac{1 - e^{-H(t)}}{1 - e^{-H(\beta)}} = \frac{1 - e^{-(t-\alpha)}}{1 - e^{-(\beta-\alpha)}},
\]

and

\[
f(t) = \frac{e^{-(t-\alpha)}}{1 - e^{-(\beta-\alpha)}},
\]

which corresponds to a shifted exponential distribution, with scale parameter 1, right-truncated at \( \beta \). This suggests that the arc length of the PDF of the Uniform(\( \alpha, \beta \)) distribution, is uniquely characterised by the shifted, right-truncated exponential distribution (with scale parameter equal to 1), i.e. shifted with parameter \( \alpha \), and right-truncated with parameter \( \beta \).

6.3 Exponential Distribution

Let \( T \sim \text{Exp}(\theta), \) i.e.

\[
g(t) = \theta e^{-\theta t}, \quad t \geq 0
\]
so that the arc length on any arbitrary interval \([a, b] \in \mathbb{R}^+\) is given by

\[
S_g^{[a, b]} = \int_a^b \sqrt{1 + (g'(x))^2} \, dx
= \int_a^b \sqrt{1 + \theta^2 (g(x))^2} \, dx.
\]

Making the substitution

\[
\theta e^{-\theta t} = \sinh u
\]

\[
\therefore u = \arcsinh(\theta e^{-\theta t})
= \ln(\theta e^{-\theta t} + \sqrt{\theta e^{-2\theta t} + 1})
\]

\[
\therefore a^* = \ln(\theta e^{-\theta a} + \sqrt{\theta e^{-2\theta a} + 1})
\]

\[
\therefore b^* = \ln(\theta e^{-\theta b} + \sqrt{\theta e^{-2\theta b} + 1})
\]

\[
\therefore -\theta^2 e^{-\theta t} \, dt = \cosh u \, du
\]

\[
\therefore \, dt = -\frac{1}{\theta} \coth u \, du.
\]

so that

\[
S_g^{[a, b]} = \int_a^b \sqrt{1 + \theta^2 e^{-2\theta t}} \, dt
= -\frac{1}{\theta} \int_{a^*}^{b^*} \sqrt{1 + \sinh^2 u} \coth u \, du
= -\frac{1}{\theta} \int_{a^*}^{b^*} \frac{\cosh^2 u}{\sinh u} \, du
= -\frac{1}{\theta} \cosh u a^* - \frac{1}{\theta} \int_{a^*}^{b^*} \csc h \, du
= -\frac{1}{\theta} \left[ \cosh u + \ln \left| \tanh \frac{u}{2} \right| \right]_{a^*}^{b^*}, \ u \neq 0.
\]

Although solvable exactly, a unrecognisable distribution is obtained, rendering any further analyses unuseable (see the preface to McElreath (2016) [58]).
6.4 Summary

This chapter provided the mathematical premises for the arc length based measures proposed in this thesis. It was seen that the arc length transformation may be viewed as a type of generating function, which implies that a distribution is characterised by an infinite collection of arc lengths, akin to other generators, such as moments.
Chapter 7

Conclusions

The conclusions of the work presented in this thesis, are summarised in this chapter, specifically, in Section 7.1. Future directions that could be taken by related research, are outlined in Section 7.2.

7.1 Summary of Conclusions

The arc length measure has provided the fuel for contemplation for some of the most basic notions encountered in Statistics. These included estimation of univariate distributions, linear regression, and non-linear regression. As statistical inference is the goal of these practises, this was highlighted throughout.

The method of arc lengths, for parameter estimation, was introduced, and it was shown that although it being consistent, generally leads to parameter estimates with greater bias and rMSE than ML, CvM, and AD methods. A discrete and continuous sample arc length statistic were proposed. In particular, the discrete arc length sample statistic, \( f(q_{0.01}, q_{0.99}) \) proved to be particularly effective for estimating \( \sigma \), and offers results comparable to that obtained by the CvM method. These arc length sample statistics provide great flexibility through its tuning parameters.

The arc length (GoF) test was constructed using the continuous sample arc length statistic, for which a parametric bootstrap yielded a null distribution. A GoF test constructed using particular tuning parameters, with parameters estimated using the discrete sample arc length statistic, yielded the most powerful tests. It was shown that the arc length test is more powerful against positively skewed and platykurtic alternatives.
than both the CvM and AD tests.

Two new regression techniques were proposed based on equating properties of KDE’s, based on a set of dependent and independent observations. Both these methods didn’t tamper with the linear model configuration, but instead proposed new objective functions. Standard errors, confidence intervals and $p$-values were obtained using the bootstrap, and the results obtained compared to that obtained from OLS regression.

It was seen that although OLS regression resulted in the smallest residual standard errors, ALR could predict a larger percentage explained variation as measured by $r^2$, but that the MMM, had the smallest divergence between the observed, and predicted value distributions.

As an alternative to NLS, and as a generalisation of ALR, NALR was introduced. The (truncated) four-parameter kappa CDF was considered as the non-linear function to be fitted, under a set of constraints. From the results obtained, it was clear that circumstances exist under which NLS regression could be improved upon, and where NALR resulted in a plausible alternative.

In summary: Computationally intensive alternatives for central techniques used in Statistics, have been developed based on the arc length of particular statistical functions. In some cases, traditional methodologies could be improved upon, using predefined optimality criterion. All the above clearly indicate that Statistics has indeed a role to play in the development of computationally intensive procedures, and that such methods may very well outperform existing techniques.

7.2 Future Work

For the techniques developed in Chapters 3 and 4, the Gaussian KDE, along with Silverman’s (rule of thumb) bandwidth selection method, were used. A natural extension is to change any of these two components, and to compare the results with established techniques in the literature, or to improve on the work presented here.

The method of arc lengths for parameter estimation and the arc length test for GoF in Chapter 3 were developed on the PDF, but could be extended to the CDF, or QF. The latter being important for distributions not possessing a PDF or CDF in closed form. However, the PDF naturally leads to multivariate generalisations of the method of arc lengths to the “method of surface areas”, and arc length test as the “surface area
test”, as the theory for multivariate KDE’s are well developed, and since the probability integral transform is not relied upon, as many of the spacings based methods, reviewed in Chapter 1, do. The normal distribution served as the vehicle for illustration of the proposed techniques, but could be applied to any other univariate probability distribution for which the derivative of the PDF can be calculated.

Although the techniques developed in this thesis were made with the implicit assumption of unlimited computational capacity, any effort made in speeding up calculations, would be advantageous. In this regard, the work done by Floater [30], and Floter and Rasmussen [31] could for instance be used, as the calculation of the arc length has been a long standing problem in numerical mathematics. By recognising the systems of equations required to be solved in Chapter 4 for the moment matching and arc length regression techniques, as systems of Fredholm equations, the work done by Twomey (1963) [81] could also yield faster, or approximations to explicit solutions.

The NALR technique, applied to the four-parameter kappa distribution in Chapter 5, could be applied to any non-linear regression problem, with or without constraints.

Alas, the resounding conclusion is that a great many work on the subjects touched upon in this thesis, will remain unconquered, and that Statistics has yet a role to play in the current age of computation.
Bibliography


Appendix A

The alR Package

Highlights of the "alR" package [54] developed for the application of the methods proposed in this thesis, are presented in this appendix.

A.1 Estimation

A.1.1 Method of Arc Lengths

\[ \text{alE} \]

*Arc length estimation.*

**Description**

A framework for arc length estimation.

**Usage**

\[
\begin{align*}
\text{alE}(x, q1, q2, dc, \text{type}) \\
\text{alEfitdist}(x, q1, q2, dc, \text{type}, \text{bootstraps}) \\
\text{alEfit}(X, q1, q2, dc, \text{type}, \text{bootstraps}, \ldots)
\end{align*}
\]

## Default S3 method:
alEfit(X, q1, q2, dc, type, bootstraps, ...)

## S3 method for class 'alEfit'
print(x, ...)

Arguments

- **x** An alEfit object.
- **q1, q2** Vectors specifying the quantiles over which arc length segments are to be computed.
- **dc** TRUE/FALSE: Should the discrete or continuous sample statistic be used.
- **type** The type of bandwidth estimator for the underlying KDE; see `bw`.
- **bootstraps** An integer specifying the size of the parametric bootstrap.
- **X** A vector of sample values.
- **...** Additional arguments passed to `alEfit` (not currently used).

Details

- Estimate distributional parameters using the method of arc lengths.
- Simulate bootstrap distributions for parameter estimates, resulting from sample arc length statistics.

This method is currently only implemented for the normal distribution. The underlying C code for the Nelder-Mead method of the optim function is used for optimising the objective function. The tolerance level is set at 1e-15, and a maximum number of 1000 iterations is allowed. The maximum likelihood estimates are used as initial values for the Nelder-Mead algorithm.

Value

alE: A list with the following components (see `optim`):

- **par**: The estimated parameters.
- **abstol**: The absolute tolerance level (default 1e-15).
- fail: An integer code indicating convergence.
- fncount: Number of function evaluations.

alEfitdist: A matrix of parameter estimates resulting from the estimated arc lengths over the specified interval(s), i.e. the bootstrap distribution for the estimated parameters resulting from the chosen sample arc length statistic.

alEfit: A generic S3 object with class alEfit.

alEfit.default: A list with all components from alE, as well as:
- dc: TRUE/FALSE Was the discrete or continuous sample arc length statistic used?
- q1, q2: The segments over which the arc length(s) were calculated.
- bw: The bandwidth for the kernel density estimator.
- dist: A numeric matrix whose columns represent a bootstrap distribution for the corresponding parameter estimate.
- se: A numeric vector with standard errors, obtained by a parametric bootstrap.
- bootstraps: Number of bootstrap samples.

Methods (by class)
- default: default method for alEfit.
- alEfit: print method for alEfit.

Examples

```r
x <- rnorm(1000)
alE(x, 0.025, 0.975, TRUE, -1)
alE(x, c(0.025, 0.5), c(0.5, 0.975), TRUE, -1)
alE(x, 0.025, 0.975, FALSE, -1)
alE(x, c(0.025, 0.5), c(0.5, 0.975), FALSE, -1)
```

```r
## Not run:
alEfitdist(x, 0.025, 0.975, TRUE, -1, 100)
alEfitdist(x, 0.025, 0.975, FALSE, -1, 100)
```
A.1.2 Arc Length Test

alEdist

Arc length estimation.

Description

Goodness-of-fit using arc lengths.

Usage

alEdist(n, bootstraps, mu, sigma, q1, q2, quantile, dc, type)

alEtest(X, mu, sigma, q1, q2, type, bootstraps, ...)

## Default S3 method:
alEtest(X, mu, sigma, q1, q2, type, bootstraps, ...)

## S3 method for class 'alEtest''
print(x, ...)

Arguments

n
An integer specifying the sample size.

bootstraps
An integer specifying the size of the parametric bootstrap.

mu
A real value specifying the mean of the normal distribution.

sigma
A positive real number specifying the scale parameter of the normal distribution.
q1, q2  Vectors specifying the quantiles (or points if `quantile=FALSE`) over which arc length segments are to be computed.

quantile  TRUE/FALSE whether q1 and q2 are quantiles, or elements of the domain of x.

dc  TRUE/FALSE: Should the discrete or continuous sample statistic be used.

type  The type of bandwidth estimator for the underlying KDE; see `bw`.

x  A vector of sample values.

...  Additional arguments passed to `alEtest` (not currently used).

x  An alEtest object.

Details

First the distributional parameters of a sample is estimated using the continuous arc length sample statistic (see `alE`). The calculated sample arc length statistic is then compared to the distribution of that particular sample statistic, obtained by a parametric bootstrap, using the estimated parameters (see `alEdist`). This finally leads to the calculation of a p-value for a goodness-of-fit test, based on the simulated distribution.

This method is currently only implemented for the normal distribution, and for a single arc length segment.

Value

alEdist: A vector (matrix) of arc lengths over the specified interval(s), i.e. the simulated distribution for the chosen sample arc length statistic.

alEtest: A generic S3 object with class alEtest.

alEtest.default: A list with the following components:

- q1, q2: The segment over which the arc length was calculated.
- mu: A real value specifying the mean of the normal distribution.
- sigma: A positive real number specifying the scale parameter of the normal distribution.
- bw: The bandwidth for the kernel density estimator.
- dist: A numeric matrix whose columns represent a bootstrap distribution for the corresponding sample arc length statistic.
- statistic: The value of the observed sample statistic.
- pvalue: The p-value for the test based on a parametric bootstrap sample.
- bootstraps: Number of bootstrap samples.

Methods (by class)

Examples

```r
## Not run:
alEdist(50, 100, 2, 3.5, 0.025, 0.975, TRUE, TRUE, -1)
alEdist(50, 100, 2, 3.5, c(0.025, 0.5), c(0.5, 0.975), TRUE, TRUE, -1)
alEdist(50, 100, 2, 3.5, 0.025, 0.975, TRUE, FALSE, -1)
alEdist(50, 100, 2, 3.5, c(0.025, 0.5), c(0.5, 0.975), TRUE, FALSE, -1)
alEdist(50, 100, 2, 3.5, qnorm(0.025, 2, 3.5), qnorm(0.975, 2, 3.5), FALSE, FALSE, -1)
alEdist(50, 100, 2, 3.5, c(qnorm(0.025, 2, 3.5), 2), c(2, qnorm(0.975, 2, 3.5)), FALSE, FALSE, -1)
## End(Not run)
## Not run:
x <- rnorm(1000)
s1 <- alE(x, 0.025, 0.975, TRUE, -1)
alEtest(x, mu=s1$par[1], sigma=s1$par[2], q1=0.025, q2=0.975, type=-1, bootstraps=50)
s2 <- alE(x, 0.025, 0.975, FALSE, -1)
alEtest(x, mu=s2$par[1], sigma=s2$par[2], q1=0.025, q2=0.975, type=-1, bootstraps=50)
## End(Not run)
```
A.2 Arc Length Regression

A.2.1 Moment Matching

\textit{mmKDEboot} \hspace{1cm} \textit{Moment matching for kernel density estimators.}

\textbf{Description}

Bootstrap estimates, along with standard errors and confidence intervals, of a linear
model, resulting from moment matching of kernel density estimates.

\textbf{Usage}

\begin{verbatim}
mmKDEboot(formula, data = list(), xin, type, bootstraps, bootName, ...)

## Default S3 method:
m.addElementClassEnabledS3Method()

## S3 method for class 'mmKDEboot'
print(x, ...)

## S3 method for class 'summary.mmKDEboot'
print(x, ...)

## S3 method for class 'formula'
print(x, ...)
\end{verbatim}

© University of Pretoria
## S3 method for class 'mmKDEboot'

```r
predict(object, newdata = NULL, ...)
```

### Arguments

- `formula`: An LHS `~` RHS formula, specifying the linear model to be estimated.
- `data`: A data.frame which contains the variables in `formula`.
- `xin`: Numeric vector of length equal to the number of independent variables, of initial values, for the parameters to be estimated.
- `type`: An integer specifying the bandwidth selection method used, see `bw`.
- `bootstraps`: An integer giving the number of bootstrap samples.
- `bootName`: The name of the .rds file to store the mmKDEboot object. May include a path.
- `...`: Arguments to be passed on to the control argument of the `optim` function.
- `x`: An mmKDEboot object.
- `object`: An mmKDEboot object.
- `newdata`: The data on which the estimated model is to be fitted.

### Value

A generic S3 object with class mmKDEboot.

`mmKDEboot.default`: A list object (saved using `saveRDS` in the specified location) with the following components:

- `intercept`: Did the model contain an intercept TRUE/FALSE?
- `coefficients`: A vector of estimated coefficients.
- `coefDist`: The bootstrap parameter distribution.
- `bcoefficients`: A vector of bootstrap coefficients, resulting from bootstrap estimation.
- `df`: Degrees of freedom of the model.
- se: The standard errors for the estimates resulting from bootstrap estimation.
- error: The value of the objective function.
- errorList: A vector of values of the objective function for each bootstrap sample.
- fitted.values: A vector of estimated values.
- residuals: The residuals resulting from the fitted model.
- call: The call to the function.
- h_y: The KDE bandwidth estimator for the dependent variable.
- h_X: The KDE bandwidth estimator for the independent variables, i.e. $X\hat{\beta}$.
- MOMy: The first $n$ non central moments of the dependent variable, where $n$ is the number of columns in the design matrix.
- MOMX: The first $n$ non central moments of the independent variables $X\hat{\beta}$, where $n$ is the number of columns in the design matrix.
- time: Min, mean and max time incurred by the computation, as obtained from comm.timer.

summary.mmKDEboot: A list of class summary.mmKDEboot with the following components:

- call: Original call to mmKDEboot function.
- coefficients: A matrix with estimates, estimated errors, and 95% parameter confidence intervals (based on the inverse empirical distribution function).
- moments: A matrix of the first $n$ moments of the dependent and independent variables that were matched. The final row corresponds to the estimated bandwidth parameters for each, i.e. $h_y$ and $h_X$, respectively.
- r.squared: The $r^2$ coefficient.
- adj.r.squared: The adjusted $r^2$ coefficient.
- sigma: The residual standard error.
- df: Degrees of freedom for the model.
- error: Value of the objective function.
- time: Min, mean and max time incurred by the computation, as obtained from `comm.timer`.
- residSum: Summary statistics for the distribution of the residuals.
- errorSum: Summary statistics for the distribution of the value of the objective function.

`print.summary.mmKDEboot`: The object passed to the function is returned invisibly.

`predict.mmKDEboot`: A vector of predicted values resulting from the estimated model.

### Methods (by class)

- `default`: default method for `mmKDEboot`.
- `mmKDEboot`: print method for `mmKDEboot`.
- `summary.mmKDEboot`: summary method for `mmKDEboot`.
- `summary.mmKDEboot`: print method for `summary.mmKDEboot`.
- `formula`: formula method for `mmKDEboot`.
- `predict.mmKDEboot`: predict method for `mmKDEboot`.

### A.2.2 Arc Length Matching

**alKDEboot**  
*Arc length matching for kernel density estimators.*

**Description**

Bootstrap estimates, along with standard errors and confidence intervals, of a linear model, resulting from arc length matching of kernel density estimates.
Usage

alkDExboot(formula, data = list(), xin, q1, q2, type, bootstraps, bootName, ...)

## Default S3 method:
alkDExboot(formula, data = list(), xin, q1, q2, type, bootstraps, bootName, ...)

## S3 method for class 'alkDExboot'
print(x, ...)

## S3 method for class 'alkDExboot'
summary(object, ...)

## S3 method for class 'summary.alkDExboot'
print(x, ...)

## S3 method for class 'formula'
alkDExboot(formula, data = list(), xin, q1, q2, type, bootstraps, bootName, ...)

## S3 method for class 'alkDExboot'
predict(object, newdata = NULL, ...)

Arguments

formula  An LHS ~ RHS formula, specifying the linear model to be estimated.
data     A data.frame which contains the variables in formula.
xin      Numeric vector of length equal to the number of independent variables, of initial values, for the parameters to be estimated.
q1, q2   Numeric vectors, for the lower and upper bounds of the intervals over which arc lengths are to be computed.
type     An integer specifying the bandwidth selection method used, see bw.
**bootstraps**  An integer giving the number of bootstrap samples.

**bootName**  The name of the .rds file to store the alKDEboot object. May include a path.

...  Arguments to be passed on to the control argument of the `optim` function.

**x**  An alKDEboot object.

**object**  An alKDEboot object.

**newdata**  The data on which the estimated model is to be fitted.

## Value

A generic S3 object with class alKDEboot.

alKDEboot.default: A list object (saved using `saveRDS` in the specified location) with the following components:

- `intercept`: Did the model contain an intercept TRUE/FALSE?
- `coefficients`: A vector of estimated coefficients.
- `coefDist`: The bootstrap parameter distribution.
- `bcoefficients`: A vector of bootstrap coefficients, resulting from bootstrap estimation.
- `df`: Degrees of freedom of the model.
- `se`: The standard errors for the estimates resulting from bootstrap estimation.
- `error`: The value of the objective function.
- `errorList`: A vector of values of the objective function for each bootstrap sample.
- `fitted.values`: A vector of estimated values.
- `residuals`: The residuals resulting from the fitted model.
- `call`: The call to the function.
- `h_y`: The KDE bandwidth estimator for the dependent variable.
- `h_X`: The KDE bandwidth estimator for the independent variables, i.e. $X \hat{\beta}$.
- `ALy`: Arc length segments of the KDE cast over the dependent variable.
- ALx: Arc length segments of the KDE cast over the independent variables $X^{\hat{\beta}}$.
  
  p1: The vector of quantiles in the domain of $y$ corresponding to $q1$. p2: The vector of quantiles in the domain of $y$ corresponding to $q2$.
  
- time: Min, mean and max time incurred by the computation, as obtained from `comm.timer`.
  
summary.alKDEboot: A list of class summary.alKDEboot with the following components:
  
- call: Original call to the `alKDEboot` function.
  
- coefficients: A matrix with estimates, estimated errors, and 95% parameter confidence intervals (based on the inverse empirical distribution function).
  
- arclengths: A matrix of the arc length segments that were matched, for the dependent and independent variables. The final row corresponds to the estimated bandwidth parameters for each, i.e. $h_y$ and $h_x$, respectively.
  
- r.squared: The $r^2$ coefficient.
  
- adj.r.squared: The adjusted $r^2$ coefficient.
  
- sigma: The residual standard error.
  
- df: Degrees of freedom for the model.
  
- error: Value of the objective function.
  
- time: Min, mean and max time incurred by the computation, as obtained from `comm.timer`.
  
- residSum: Summary statistics for the distribution of the residuals.
  
- errorSum: Summary statistics for the distribution of the value of the objective function.
  
print.summary.alKDEboot: The object passed to the function is returned invisibly.
  
predict.alKDEboot: A vector of predicted values resulting from the estimated model.

Methods (by class)
  
- default: default method for alKDEboot.
A.2.3 Bhattacharyya Divergence Test

**bhatt.test**  
*Bhattacharyya test for comparing two samples.*

**Description**

Use the multinomial distribution to test the hypothesis that two samples come from the same distribution.

**Usage**

```r
bhatt.test(y, x, k)
```

**Arguments**

- `y, x`  
  Samples to be compared.
- `k`  
  Number of proportions.

**Details**

It is assumed that the two samples come from the same kernel density distribution. The support of the KDE of the first sample is divided into $k$ equally spaced quantiles, and then compared to the corresponding proportions of the second.
Value

bat.test: A list with the following components:

- df=2*k: where k is the number of proportions used.
- y.prop, x.prop: Vectors of proportions.
- D2: Measure of divergence between samples.
- test.stat: The test statistic for the Bhattacharyya test.
- p.value: The p-value of the test.

Examples

```r
y <- rnorm(1000)
x <- rnorm(1000)
bhatt.test(y, x, 10)
```

A.3 Non-Linear Arc Length Regression

A.3.1 Non-linear Least Squares

kappa4nlsBoot

_Sigmoidal curve fitting._

Description

Bootstrap estimates, along with standard errors and confidence intervals, of a non-linear model, resulting from nonlinear least squares fitting of the four-parameter kappa sigmoidal function.

Usage

```r
kappa4nlsBoot(formula, data = list(), xin, lower, upper, tol, maxiter,
               bootstraps, bootName, ...)
```
## Default S3 method:
kappa4nlsBoot(formula, data = list(), xin, lower = c(0, 
-5, -5), upper = c(10, 1, 1), tol = 1e-15, maxiter = 50000, bootstraps,
bootName, ...)

## S3 method for class 'kappa4nlsBoot'
print(x, ...)

## S3 method for class 'kappa4nlsBoot'
summary(object, ...)

## S3 method for class 'summary.kappa4nlsBoot'
print(x, ...)

## S3 method for class 'formula'
kappa4nlsBoot(formula, data = list(), xin, lower, upper,
   tol, maxiter, bootstraps, bootName, ...)

## S3 method for class 'kappa4nlsBoot'
predict(object, newdata = NULL, ...)

### Arguments

- **formula**: An LHS ~ RHS formula, specifying the linear model to be estimated.
- **data**: A data.frame which contains the variables in formula.
- **xin**: Numeric vector of length 3 containing initial values, for \( \sigma \), \( h \), and \( k \).
- **lower**: A vector of lower constraints for the parameters to be estimated; defaults to \( c(0, -5, -5) \).
- **upper**: A vector of upper constraints for the parameters to be estimated; defaults to \( c(10, 1, 1) \).
- **tol**: Error tolerance level; defaults to 1e-15.
- **maxiter**: The maximum number of iterations allowed; defaults to 50000.
bootstraps  An integer giving the number of bootstrap samples.

bootName  The name of the .rds file to store the kappa4nlsBoot object. May include a path.

...  Arguments to be passed on to the differential evolution function JDEoptim.

x  A kappa4nlsBoot object.

object  A kappa4nlsBoot object.

newdata  The data on which the estimated model is to be fitted.

Value

A generic S3 object with class kappa4nlsBoot.

kappa4nlsBoot.default: A list object (saved using saveRDS in the specified location) with the following components:

- intercept: Did the model contain an intercept TRUE/FALSE?
- coefficients: A vector of estimated coefficients.
- bcoefficients: A vector of bootstrap coefficients, resulting from bootstrap estimation.
- se: The standard errors for the estimates resulting from bootstrap estimation.
- error: The value of the objective function.
- errorList: A vector of values of the objective function for each bootstrap sample.
- fitted.values: A vector of estimated values.
- residuals: The residuals resulting from the fitted model.
- call: The call to the function.
- time: Min, mean and max time incurred by the computation, as obtained from comm.timer.

summary.kappa4nlsBoot: A list of class summary.kappa4nlsBoot with the following components:

- call: Original call to the kappa4nlsBoot function.
coefficients: A matrix with estimates, estimated errors, and 95% parameter confidence intervals (based on the inverse empirical distribution function).

r.squared: The $r^2$ coefficient.

sigma: The residual standard error.

error: Value of the objective function.

time: Min, mean and max time incurred by the computation, as obtained from \texttt{comm.timer}.

residSum: Summary statistics for the distribution of the residuals.

errorSum: Summary statistics for the distribution of the value of the objective function.

\texttt{print.summary.kappa4nlsBoot}: The object passed to the function is returned invisibly.

\texttt{predict.kappa4nlsBoot}: A vector of predicted values resulting from the estimated model.

Methods (by class)

- \texttt{default}: default method for \texttt{kappa4nlsBoot}.
- \texttt{kappa4nlsBoot}: print method for \texttt{kappa4nlsBoot}.
- \texttt{kappa4nlsBoot}: summary method for \texttt{kappa4nlsBoot}.
- \texttt{summary.kappa4nlsBoot}: print method for \texttt{summary.kappa4nlsBoot}.
- \texttt{formula}: formula method for \texttt{kappa4nlsBoot}.
- \texttt{kappa4nlsBoot}: predict method for \texttt{kappa4nlsBoot}.

\textbf{A.3.2 Non-Linear Arc Length Regression}
**kappa4alBoot**

*Sigmoidal curve fitting.*

**Description**

Bootstrap estimates, along with standard errors and confidence intervals, of a non-linear model, resulting from arc length fitting of the four-parameter kappa sigmoidal function.

**Usage**

```r
kappa4alBoot(formula, data = list(), xin, lower, upper, q1, q2, tol, maxiter,
  bootstraps, bootName, ...)

# Default S3 method:
kappa4alBoot(formula, data = list(), xin, lower = c(0, -5, -5), upper = c(10, 1, 1), q1, q2, tol = 1e-15, maxiter = 50000,
  bootstraps, bootName, ...)

# S3 method for class 'kappa4alBoot'
print(x, ...)

# S3 method for class 'kappa4alBoot'
summary(object, ...)

# S3 method for class 'summary.kappa4alBoot'
print(x, ...)

# S3 method for class 'formula'
kappa4alBoot(formula, data = list(), xin, lower, upper, q1, q2, tol, maxiter, bootstraps, bootName, ...)

# S3 method for class 'kappa4alBoot'
predict(object, newdata = NULL, ...)
```
Arguments

formula An LHS ~ RHS formula, specifying the linear model to be estimated.
data A data.frame which contains the variables in formula.
xin Numeric vector of length 3 containing initial values, for \( \sigma, h \), and \( k \).
lower A vector of lower constraints for the parameters to be estimated; defaults to \( c(0, -5, -5) \).
upper A vector of upper constraints for the parameters to be estimated; defaults to \( c(10, 1, 1) \).
q1, q2 Numeric vectors, for the lower and upper bounds of the intervals over which arc lengths are to be computed.
tol Error tolerance level; defaults to 1e-15.
maxiter The maximum number of iterations allowed; defaults to 50000.
bootstraps An integer giving the number of bootstrap samples.
bootName The name of the .rds file to store the kappa4alBoot object. May include a path.
... Arguments to be passed on to the differential evolution function JDEoptim.
x A kappa4alBoot object.
object A kappa4alBoot object.
newdata The data on which the estimated model is to be fitted.

Value

A generic S3 object with class kappa4alBoot.
kappa4alBoot.default: A list object (saved using saveRDS in the specified location) with the following components:

- intercept: Did the model contain an intercept TRUE/FALSE?
- coefficients: A vector of estimated coefficients.
- bcoefficients: A vector of bootstrap coefficients, resulting from bootstrap estimation.
- se: The standard errors for the estimates resulting from bootstrap estimation.
- error: The value of the objective function.
- errorList: A vector of values of the objective function for each bootstrap sample.
- fitted.values: A vector of estimated values.
- residuals: The residuals resulting from the fitted model.
- call: The call to the function.
- time: Min, mean and max time incurred by the computation, as obtained from `comm.timer`.

`summary.kappa4alBoot`: A list of class `summary.kappa4alBoot` with the following components:

- call: Original call to the `kappa4alBoot` function.
- coefficients: A matrix with estimates, estimated errors, and 95% parameter confidence intervals (based on the inverse empirical distribution function).
- arclengths: A matrix of the arc length segments that were matched, for the dependent and independent variables.
- r.squared: The $r^2$ coefficient.
- sigma: The residual standard error.
- error: Value of the objective function.
- time: Min, mean and max time incurred by the computation, as obtained from `comm.timer`.
- residSum: Summary statistics for the distribution of the residuals.
- errorSum: Summary statistics for the distribution of the value of the objective function.

`print.summary.kappa4alBoot`: The object passed to the function is returned invisibly.

`predict.kappa4alBoot`: A vector of predicted values resulting from the estimated model.
Methods (by class)

- **default**: default method for `kappa4alBoot`.
- **kappa4alBoot**: print method for `kappa4alBoot`.
- **kappa4alBoot**: summary method for `kappa4alBoot`.
- **summary.kappa4alBoot**: print method for `summary.kappa4alBoot`.
- **formula**: formula method for `kappa4alBoot`.
- **kappa4alBoot**: predict method for `kappa4alBoot`.

### A.4 Summary

Key functions and methods from the alR package were presented in this appendix. The source code and complete package documentation are available at [https://github.com/mtloots/alR](https://github.com/mtloots/alR).
Appendix B

Simulation Results for the Method of Arc Lengths

In this Appendix, simulation results for the method of arc lengths, applied to the normal distribution, are presented. Here, the results presented in Chapter 3, Section 3.3 are extended to include a larger number of parameter choices for the normal distribution. For each of the resulting parameter choices (μ and σ) of the normal distribution, results for the estimated bias and rMSE values (in ascending order of rMSE) for the median values, of the parameters estimated from 1,000 samples of size 1,000, are given. Performance results are also provided for all these (μ, σ) combinations, by varying the sample size. The estimation techniques to be compared correspond to those outlined in Section 3.3, but is reproduced in Section B.1 for completeness. The remaining sections correspond to a particular (μ, σ) combination.

B.1 Introduction

The method of arc lengths (resulting from the two proposed sample arc length statistics given in (3.2.1) and (3.2.2)), ML, CvM, and AD estimation techniques are compared. For each of the proposed sample arc length statistics, five variants were included depending on the interval [a, b], where the two end points are specified in terms of the sample quantiles:

(a) [0.01, 0.99],
Appendix B. Simulation Results for the Method of Arc Lengths

(b) \([0.025, 0.975]\),

(c) \([0.05, 0.95]\),

(d) \([0.075, 0.925]\), and

(e) \([0.1, 0.9]\).

B.2 Simulation Results for \(N(-2, 0.5)\)

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00006</td>
<td>0.01072</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00025</td>
<td>0.01087</td>
</tr>
<tr>
<td>CvM</td>
<td>0.00066</td>
<td>0.01114</td>
</tr>
<tr>
<td>s1b</td>
<td>0.01763</td>
<td>0.03340</td>
</tr>
<tr>
<td>s1d</td>
<td>0.03984</td>
<td>0.04221</td>
</tr>
<tr>
<td>s1c</td>
<td>0.02891</td>
<td>0.04503</td>
</tr>
<tr>
<td>s1a</td>
<td>0.05921</td>
<td>0.07764</td>
</tr>
<tr>
<td>s1e</td>
<td>0.01506</td>
<td>0.09161</td>
</tr>
<tr>
<td>s2c</td>
<td>0.11299</td>
<td>0.11757</td>
</tr>
<tr>
<td>s2d</td>
<td>0.11563</td>
<td>0.12153</td>
</tr>
<tr>
<td>s2b</td>
<td>0.12032</td>
<td>0.12417</td>
</tr>
<tr>
<td>s2a</td>
<td>0.13285</td>
<td>0.13442</td>
</tr>
<tr>
<td>s2e</td>
<td>0.12910</td>
<td>0.13542</td>
</tr>
</tbody>
</table>

Table B.1: Estimated bias and rMSE for \(\mu=-2\) and \(\sigma=0.5\).
Appendix B. Simulation Results for the Method of Arc Lengths

(a) Estimated rMSE for $\mu = -2$.

(b) Estimated rMSE for $\sigma = 0.5$.

Figure B.1: Estimated rMSE for $\mu = -2$ and $\sigma = 0.5$ for varying sample sizes.

© University of Pretoria
Appendix B. Simulation Results for the Method of Arc Lengths

(a) Estimated rMSE for $\mu = -2$.

(b) Estimated rMSE for $\sigma = 1$.

Figure B.2: Estimated rMSE for $\mu = -2$ and $\sigma = 1$ for varying sample sizes.
Appendix B. Simulation Results for the Method of Arc Lengths

(a) Estimated rMSE for $\mu = -2$.

(b) Estimated rMSE for $\sigma = 3.5$.

Figure B.3: Estimated rMSE for $\mu = -2$ and $\sigma = 3.5$ for varying sample sizes.
Figure B.4: Estimated rMSE for $\mu = 0$ and $\sigma = 0.5$ for varying sample sizes.
<table>
<thead>
<tr>
<th>Method</th>
<th>Bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00104</td>
<td>0.00770</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00080</td>
<td>0.00825</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00246</td>
<td>0.00911</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00155</td>
<td>0.00922</td>
</tr>
<tr>
<td>s2b</td>
<td>0.00607</td>
<td>0.01029</td>
</tr>
<tr>
<td>s2a</td>
<td>0.00714</td>
<td>0.01092</td>
</tr>
<tr>
<td>s2c</td>
<td>0.00522</td>
<td>0.01107</td>
</tr>
<tr>
<td>s2d</td>
<td>0.00551</td>
<td>0.01181</td>
</tr>
<tr>
<td>s2e</td>
<td>0.00594</td>
<td>0.01406</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.02060</td>
<td>0.02061</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.05053</td>
<td>0.05053</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.07608</td>
<td>0.07608</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.10350</td>
<td>0.10350</td>
</tr>
</tbody>
</table>

Table B.2: Estimated bias and rMSE for $\mu=-2$ and $\sigma=0.5$. 
Appendix B. Simulation Results for the Method of Arc Lengths

(a) Estimated rMSE for $\mu = 0$.

(b) Estimated rMSE for $\sigma = 1$.

Figure B.5: Estimated rMSE for $\mu = 0$ and $\sigma = 1$ for varying sample sizes.

© University of Pretoria
Appendix B. Simulation Results for the Method of Arc Lengths

Figure B.6: Estimated rMSE for $\mu=0$ and $\sigma=3.5$ for varying sample sizes.

(a) Estimated rMSE for $\mu=0$.

(b) Estimated rMSE for $\sigma=3.5$. 
Appendix B. Simulation Results for the Method of Arc Lengths

(a) Estimated rMSE for $\mu = 2$.

(b) Estimated rMSE for $\sigma = 0.5$.

Figure B.7: Estimated rMSE for $\mu = 2$ and $\sigma = 0.5$ for varying sample sizes.

© University of Pretoria
Appendix B. Simulation Results for the Method of Arc Lengths

(a) Estimated rMSE for $\mu = 2$.

(b) Estimated rMSE for $\sigma = 1$.

Figure B.8: Estimated rMSE for $\mu = 2$ and $\sigma = 1$ for varying sample sizes.

© University of Pretoria
Appendix B. Simulation Results for the Method of Arc Lengths

(a) Estimated rMSE for $\mu = 2$.

(b) Estimated rMSE for $\sigma = 3.5$.

Figure B.9: Estimated rMSE for $\mu = 2$ and $\sigma = 3.5$ for varying sample sizes.
### Appendix B. Simulation Results for the Method of Arc Lengths

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AD</td>
<td>0.00159</td>
<td>0.02247</td>
</tr>
<tr>
<td>MLE</td>
<td>0.00204</td>
<td>0.02254</td>
</tr>
<tr>
<td>CvM</td>
<td>0.00126</td>
<td>0.02267</td>
</tr>
<tr>
<td>s1b</td>
<td>0.02074</td>
<td>0.04976</td>
</tr>
<tr>
<td>s1e</td>
<td>0.04274</td>
<td>0.06001</td>
</tr>
<tr>
<td>s1c</td>
<td>0.02995</td>
<td>0.08081</td>
</tr>
<tr>
<td>s1d</td>
<td>0.09801</td>
<td>0.09832</td>
</tr>
<tr>
<td>s1a</td>
<td>0.08571</td>
<td>0.10128</td>
</tr>
<tr>
<td>s2d</td>
<td>0.14946</td>
<td>0.15240</td>
</tr>
<tr>
<td>s2e</td>
<td>0.15296</td>
<td>0.15386</td>
</tr>
<tr>
<td>s2c</td>
<td>0.15346</td>
<td>0.15669</td>
</tr>
<tr>
<td>s2b</td>
<td>0.16216</td>
<td>0.16242</td>
</tr>
<tr>
<td>s2a</td>
<td>0.16603</td>
<td>0.16615</td>
</tr>
</tbody>
</table>

*Table B.3:* Estimated bias and rMSE for $\mu = -2$ and $\sigma = 1$.

#### B.3 Simulation Results for $N(-2, 1)$

#### B.4 Simulation Results for $N(-2, 3.5)$

#### B.5 Simulation Results for $N(0, 0.5)$

#### B.6 Simulation Results for $N(0, 1)$

#### B.7 Simulation Results for $N(0, 3.5)$

#### B.8 Simulation Results for $N(2, 0.5)$

#### B.9 Simulation Results for $N(2, 1)$

#### B.10 Simulation Results for $N(2, 3.5)$

#### B.11 Summary

Using the method of arc lengths with $\int_{q_{0.01}}^{q_{0.99}}$ yielded the smallest bias and rMSE values for simultaneously estimating $\mu$ and $\sigma$. All sample arc length statistics performed
Appendix B. Simulation Results for the Method of Arc Lengths

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>0.00091</td>
<td>0.01584</td>
</tr>
<tr>
<td>AD</td>
<td>0.00060</td>
<td>0.01660</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00093</td>
<td>0.01832</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00083</td>
<td>0.01874</td>
</tr>
<tr>
<td>s2a</td>
<td>0.01896</td>
<td>0.02392</td>
</tr>
<tr>
<td>s2b</td>
<td>0.01891</td>
<td>0.02525</td>
</tr>
<tr>
<td>s2c</td>
<td>0.01899</td>
<td>0.02609</td>
</tr>
<tr>
<td>s2d</td>
<td>0.01970</td>
<td>0.02691</td>
</tr>
<tr>
<td>s2e</td>
<td>0.01960</td>
<td>0.02849</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.03849</td>
<td>0.03860</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.10036</td>
<td>0.10036</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.16021</td>
<td>0.16021</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.21289</td>
<td>0.21289</td>
</tr>
</tbody>
</table>

Table B.4: Estimated bias and rMSE for $\mu=-2$ and $\sigma=1$.

The rMSE values for the method of arc lengths with $\int_{q_{0.01}}^{q_{0.99}}$ decrease for increasing sample sizes. For estimating $\sigma$, this corresponds well to the values obtained for the CvM method, but are not as strong as that obtained for the MLE and AD methods.
Appendix B. Simulation Results for the Method of Arc Lengths

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>0.01087</td>
<td>0.07410</td>
</tr>
<tr>
<td>AD</td>
<td>0.01130</td>
<td>0.07786</td>
</tr>
<tr>
<td>CvM</td>
<td>0.01288</td>
<td>0.07813</td>
</tr>
<tr>
<td>s1b</td>
<td>0.10162</td>
<td>0.14077</td>
</tr>
<tr>
<td>s1c</td>
<td>0.14525</td>
<td>0.18590</td>
</tr>
<tr>
<td>s1a</td>
<td>0.19016</td>
<td>0.20837</td>
</tr>
<tr>
<td>s2e</td>
<td>0.23234</td>
<td>0.24271</td>
</tr>
<tr>
<td>s2d</td>
<td>0.24803</td>
<td>0.25292</td>
</tr>
<tr>
<td>s2a</td>
<td>0.25027</td>
<td>0.25466</td>
</tr>
<tr>
<td>s2c</td>
<td>0.25603</td>
<td>0.25810</td>
</tr>
<tr>
<td>s2b</td>
<td>0.26368</td>
<td>0.26918</td>
</tr>
<tr>
<td>s1e</td>
<td>0.28161</td>
<td>0.28161</td>
</tr>
<tr>
<td>s1d</td>
<td>0.28937</td>
<td>0.29609</td>
</tr>
</tbody>
</table>

Table B.5: Estimated bias and rMSE for $\mu=-2$ and $\sigma=3.5$.

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00283</td>
<td>0.05254</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00155</td>
<td>0.05966</td>
</tr>
<tr>
<td>CvM</td>
<td>0.00061</td>
<td>0.06765</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00776</td>
<td>0.06851</td>
</tr>
<tr>
<td>s2a</td>
<td>0.06200</td>
<td>0.08184</td>
</tr>
<tr>
<td>s2b</td>
<td>0.06885</td>
<td>0.08602</td>
</tr>
<tr>
<td>s2c</td>
<td>0.07539</td>
<td>0.09110</td>
</tr>
<tr>
<td>s2d</td>
<td>0.08150</td>
<td>0.10362</td>
</tr>
<tr>
<td>s2e</td>
<td>0.08922</td>
<td>0.11406</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.13474</td>
<td>0.13521</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.35352</td>
<td>0.35352</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.56281</td>
<td>0.56281</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.74744</td>
<td>0.74744</td>
</tr>
</tbody>
</table>

Table B.6: Estimated bias and rMSE for $\mu=-2$ and $\sigma=3.5$. 

© University of Pretoria
### Appendix B. Simulation Results for the Method of Arc Lengths

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AD</td>
<td>-0.00007</td>
<td>0.00948</td>
</tr>
<tr>
<td>MLE</td>
<td>0.00015</td>
<td>0.00974</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00029</td>
<td>0.00977</td>
</tr>
<tr>
<td>s1b</td>
<td>0.01305</td>
<td>0.01927</td>
</tr>
<tr>
<td>s1c</td>
<td>0.02132</td>
<td>0.02557</td>
</tr>
<tr>
<td>s1a</td>
<td>0.02436</td>
<td>0.02745</td>
</tr>
<tr>
<td>s1d</td>
<td>0.02261</td>
<td>0.03087</td>
</tr>
<tr>
<td>s2a</td>
<td>0.03344</td>
<td>0.03417</td>
</tr>
<tr>
<td>s2e</td>
<td>0.03273</td>
<td>0.03442</td>
</tr>
<tr>
<td>s2b</td>
<td>0.03426</td>
<td>0.03503</td>
</tr>
<tr>
<td>s2d</td>
<td>0.03443</td>
<td>0.03520</td>
</tr>
<tr>
<td>s2c</td>
<td>0.03507</td>
<td>0.03556</td>
</tr>
<tr>
<td>s1e</td>
<td>0.05028</td>
<td>0.05028</td>
</tr>
</tbody>
</table>

**Table B.7:** Estimated bias and rMSE for $\mu=0$ and $\sigma=0.5$.

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00016</td>
<td>0.00792</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00046</td>
<td>0.00866</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00022</td>
<td>0.00943</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00101</td>
<td>0.00982</td>
</tr>
<tr>
<td>s2a</td>
<td>0.01059</td>
<td>0.01229</td>
</tr>
<tr>
<td>s2b</td>
<td>0.01096</td>
<td>0.01278</td>
</tr>
<tr>
<td>s2c</td>
<td>0.01155</td>
<td>0.01361</td>
</tr>
<tr>
<td>s2d</td>
<td>0.01213</td>
<td>0.01460</td>
</tr>
<tr>
<td>s2e</td>
<td>0.01210</td>
<td>0.01578</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.01907</td>
<td>0.01912</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.04828</td>
<td>0.04828</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.07452</td>
<td>0.07452</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.09949</td>
<td>0.09949</td>
</tr>
</tbody>
</table>

**Table B.8:** Estimated bias and rMSE for $\mu=0$ and $\sigma=0.5$. 
### Table B.9: Estimated bias and rMSE for $\mu=0$ and $\sigma=1$. 

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>0.00097</td>
<td>0.02182</td>
</tr>
<tr>
<td>AD</td>
<td>0.00127</td>
<td>0.02188</td>
</tr>
<tr>
<td>CvM</td>
<td>0.00207</td>
<td>0.02202</td>
</tr>
<tr>
<td>s1b</td>
<td>0.02540</td>
<td>0.03881</td>
</tr>
<tr>
<td>s1c</td>
<td>0.04050</td>
<td>0.05153</td>
</tr>
<tr>
<td>s1a</td>
<td>0.05126</td>
<td>0.05509</td>
</tr>
<tr>
<td>s2e</td>
<td>0.06404</td>
<td>0.06618</td>
</tr>
<tr>
<td>s2d</td>
<td>0.06669</td>
<td>0.06785</td>
</tr>
<tr>
<td>s2a</td>
<td>0.06650</td>
<td>0.06872</td>
</tr>
<tr>
<td>s2b</td>
<td>0.06943</td>
<td>0.06992</td>
</tr>
<tr>
<td>s2c</td>
<td>0.07017</td>
<td>0.07123</td>
</tr>
<tr>
<td>s2a</td>
<td>0.08112</td>
<td>0.08136</td>
</tr>
<tr>
<td>s1d</td>
<td>0.07916</td>
<td>0.08266</td>
</tr>
</tbody>
</table>

### Table B.10: Estimated bias and rMSE for $\mu=0$ and $\sigma=1$. 

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00216</td>
<td>0.01549</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00169</td>
<td>0.01637</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00345</td>
<td>0.01839</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00204</td>
<td>0.01868</td>
</tr>
<tr>
<td>s2a</td>
<td>0.01757</td>
<td>0.02489</td>
</tr>
<tr>
<td>s2b</td>
<td>0.01858</td>
<td>0.02571</td>
</tr>
<tr>
<td>s2c</td>
<td>0.01938</td>
<td>0.02638</td>
</tr>
<tr>
<td>s2d</td>
<td>0.01899</td>
<td>0.02794</td>
</tr>
<tr>
<td>s2e</td>
<td>0.02033</td>
<td>0.02976</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.03980</td>
<td>0.03989</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.10192</td>
<td>0.10192</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.16084</td>
<td>0.16084</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.21481</td>
<td>0.21481</td>
</tr>
</tbody>
</table>
### Table B.11: Estimated bias and rMSE for $\mu=0$ and $\sigma=3.5$.

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>0.00561</td>
<td>0.08044</td>
</tr>
<tr>
<td>AD</td>
<td>0.00578</td>
<td>0.08117</td>
</tr>
<tr>
<td>CvM</td>
<td>0.00799</td>
<td>0.08486</td>
</tr>
<tr>
<td>s1b</td>
<td>0.10402</td>
<td>0.14333</td>
</tr>
<tr>
<td>s1c</td>
<td>0.14193</td>
<td>0.17495</td>
</tr>
<tr>
<td>s1a</td>
<td>0.18542</td>
<td>0.20705</td>
</tr>
<tr>
<td>s2e</td>
<td>0.23322</td>
<td>0.24246</td>
</tr>
<tr>
<td>s2d</td>
<td>0.24376</td>
<td>0.24918</td>
</tr>
<tr>
<td>s2a</td>
<td>0.24670</td>
<td>0.25151</td>
</tr>
<tr>
<td>s2b</td>
<td>0.24762</td>
<td>0.25291</td>
</tr>
<tr>
<td>s2c</td>
<td>0.25634</td>
<td>0.26252</td>
</tr>
<tr>
<td>s1e</td>
<td>0.28892</td>
<td>0.28892</td>
</tr>
<tr>
<td>s1d</td>
<td>0.29646</td>
<td>0.30524</td>
</tr>
</tbody>
</table>

### Table B.12: Estimated bias and rMSE for $\mu=0$ and $\sigma=3.5$.

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00568</td>
<td>0.05577</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00108</td>
<td>0.06063</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00237</td>
<td>0.06906</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00731</td>
<td>0.07027</td>
</tr>
<tr>
<td>s2a</td>
<td>0.06397</td>
<td>0.08520</td>
</tr>
<tr>
<td>s2b</td>
<td>0.06810</td>
<td>0.08653</td>
</tr>
<tr>
<td>s2c</td>
<td>0.07346</td>
<td>0.09464</td>
</tr>
<tr>
<td>s2d</td>
<td>0.07793</td>
<td>0.10631</td>
</tr>
<tr>
<td>s2e</td>
<td>0.08362</td>
<td>0.11173</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.13743</td>
<td>0.13824</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.35376</td>
<td>0.35376</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.56209</td>
<td>0.56209</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.75169</td>
<td>0.75169</td>
</tr>
</tbody>
</table>
### Table B.13: Estimated bias and rMSE for $\mu=2$ and $\sigma=0.5$.

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AD</td>
<td>-0.00027</td>
<td>0.01093</td>
</tr>
<tr>
<td>MLE</td>
<td>-0.00013</td>
<td>0.01107</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00059</td>
<td>0.01150</td>
</tr>
<tr>
<td>s1b</td>
<td>0.01704</td>
<td>0.03667</td>
</tr>
<tr>
<td>s1d</td>
<td>0.04165</td>
<td>0.04305</td>
</tr>
<tr>
<td>s1c</td>
<td>0.03247</td>
<td>0.04825</td>
</tr>
<tr>
<td>s1a</td>
<td>0.06260</td>
<td>0.08087</td>
</tr>
<tr>
<td>s1e</td>
<td>0.01917</td>
<td>0.08915</td>
</tr>
<tr>
<td>s2c</td>
<td>0.11502</td>
<td>0.11815</td>
</tr>
<tr>
<td>s2d</td>
<td>0.12322</td>
<td>0.12526</td>
</tr>
<tr>
<td>s2a</td>
<td>0.12209</td>
<td>0.12601</td>
</tr>
<tr>
<td>s2b</td>
<td>0.12340</td>
<td>0.12645</td>
</tr>
<tr>
<td>s2e</td>
<td>0.13347</td>
<td>0.13498</td>
</tr>
</tbody>
</table>

### Table B.14: Estimated bias and rMSE for $\mu=2$ and $\sigma=0.5$.

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00097</td>
<td>0.00731</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00064</td>
<td>0.00805</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00267</td>
<td>0.00932</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00133</td>
<td>0.00933</td>
</tr>
<tr>
<td>s2b</td>
<td>0.00563</td>
<td>0.00997</td>
</tr>
<tr>
<td>s2c</td>
<td>0.00569</td>
<td>0.01064</td>
</tr>
<tr>
<td>s2a</td>
<td>0.00694</td>
<td>0.01077</td>
</tr>
<tr>
<td>s2d</td>
<td>0.00579</td>
<td>0.01097</td>
</tr>
<tr>
<td>s2e</td>
<td>0.00651</td>
<td>0.01305</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.02075</td>
<td>0.02098</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.05072</td>
<td>0.05072</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.07583</td>
<td>0.07583</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.10301</td>
<td>0.10301</td>
</tr>
</tbody>
</table>

© University of Pretoria
Appendix B. Simulation Results for the Method of Arc Lengths

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>0.00320</td>
<td>0.02049</td>
</tr>
<tr>
<td>CvM</td>
<td>0.00359</td>
<td>0.02060</td>
</tr>
<tr>
<td>AD</td>
<td>0.00295</td>
<td>0.02087</td>
</tr>
<tr>
<td>s1b</td>
<td>0.02300</td>
<td>0.04957</td>
</tr>
<tr>
<td>s1e</td>
<td>0.03746</td>
<td>0.05971</td>
</tr>
<tr>
<td>s1c</td>
<td>0.04644</td>
<td>0.08729</td>
</tr>
<tr>
<td>s1a</td>
<td>0.08355</td>
<td>0.09824</td>
</tr>
<tr>
<td>s1d</td>
<td>0.10210</td>
<td>0.10254</td>
</tr>
<tr>
<td>s2d</td>
<td>0.14307</td>
<td>0.14375</td>
</tr>
<tr>
<td>s2e</td>
<td>0.14556</td>
<td>0.14890</td>
</tr>
<tr>
<td>s2c</td>
<td>0.15196</td>
<td>0.15399</td>
</tr>
<tr>
<td>s2b</td>
<td>0.15756</td>
<td>0.15923</td>
</tr>
<tr>
<td>s2a</td>
<td>0.16576</td>
<td>0.16606</td>
</tr>
</tbody>
</table>

Table B.15: Estimated bias and rMSE for $\mu=2$ and $\sigma=1$.

<table>
<thead>
<tr>
<th></th>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00141</td>
<td>0.01498</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00037</td>
<td>0.01605</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00124</td>
<td>0.01811</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00305</td>
<td>0.01891</td>
</tr>
<tr>
<td>s2a</td>
<td>0.01665</td>
<td>0.02246</td>
</tr>
<tr>
<td>s2b</td>
<td>0.01699</td>
<td>0.02267</td>
</tr>
<tr>
<td>s2c</td>
<td>0.01763</td>
<td>0.02393</td>
</tr>
<tr>
<td>s2d</td>
<td>0.01899</td>
<td>0.02618</td>
</tr>
<tr>
<td>s2e</td>
<td>0.01851</td>
<td>0.02921</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.04018</td>
<td>0.04037</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.10202</td>
<td>0.10202</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.16047</td>
<td>0.16047</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.21216</td>
<td>0.21216</td>
</tr>
</tbody>
</table>

Table B.16: Estimated bias and rMSE for $\mu=2$ and $\sigma=1$. 
### Appendix B. Simulation Results for the Method of Arc Lengths

<table>
<thead>
<tr>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00103</td>
</tr>
<tr>
<td>AD</td>
<td>-0.00058</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00212</td>
</tr>
<tr>
<td>s1b</td>
<td>0.09716</td>
</tr>
<tr>
<td>s1c</td>
<td>0.14438</td>
</tr>
<tr>
<td>s1a</td>
<td>0.17579</td>
</tr>
<tr>
<td>s2e</td>
<td>0.22476</td>
</tr>
<tr>
<td>s2b</td>
<td>0.23539</td>
</tr>
<tr>
<td>s2d</td>
<td>0.23676</td>
</tr>
<tr>
<td>s2a</td>
<td>0.24221</td>
</tr>
<tr>
<td>s2c</td>
<td>0.24751</td>
</tr>
<tr>
<td>s1e</td>
<td>0.27871</td>
</tr>
<tr>
<td>s1d</td>
<td>0.28460</td>
</tr>
</tbody>
</table>

**Table B.17:** Estimated bias and rMSE for $\mu=2$ and $\sigma=3.5$.

<table>
<thead>
<tr>
<th>bias</th>
<th>rMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLE</td>
<td>-0.00122</td>
</tr>
<tr>
<td>AD</td>
<td>0.00210</td>
</tr>
<tr>
<td>CvM</td>
<td>-0.00148</td>
</tr>
<tr>
<td>s1a</td>
<td>-0.00415</td>
</tr>
<tr>
<td>s2a</td>
<td>0.06506</td>
</tr>
<tr>
<td>s2b</td>
<td>0.06905</td>
</tr>
<tr>
<td>s2c</td>
<td>0.07328</td>
</tr>
<tr>
<td>s2d</td>
<td>0.07606</td>
</tr>
<tr>
<td>s2e</td>
<td>0.07942</td>
</tr>
<tr>
<td>s1b</td>
<td>-0.13569</td>
</tr>
<tr>
<td>s1c</td>
<td>-0.35565</td>
</tr>
<tr>
<td>s1d</td>
<td>-0.55801</td>
</tr>
<tr>
<td>s1e</td>
<td>-0.74934</td>
</tr>
</tbody>
</table>

**Table B.18:** Estimated bias and rMSE for $\mu=2$ and $\sigma=3.5$. © University of Pretoria
Appendix C

Machine Learning

In this Appendix, two popular “machine learning” techniques are used for analysing the data used in Chapter 4. This is done as benchmark techniques for that, newly proposed in this thesis. The random forest (RF) is applied in Section C.1, while support vector regression (SVR) follows in Section C.2.

C.1 Random Forest

The “randomForest” function from the similarly named R package by Liaw and Wiener (2002) [52] was used for fitting the RF in this section. The output is shown below:

Call:
  randomForest(formula = FPKM_lambda ~ H3K4me3_signal_bin25 +
                H3K27me3_signal_bin21, data = training, ntree = 500, xtest =
                data.frame(testing$H3K4me3_signal_bin25,
                testing$H3K27me3_signal_bin21), ytest = testing$FPKM_lambda)
Type of random forest: regression
  Number of trees: 500
No. of variables tried at each split: 1

Mean of squared residuals: 0.22259
  % Var explained: 46.9
  Test set MSE: 0.22
  % Var explained: 47.18
Figure C.1 compares the KDE’s for the dependent variable, with the predictions resulting from the RF fit. In each case, the smoothing parameter is also reported.

(a) KDE’s for transformed FPKM signal and the random forest fit thereof, with bandwidths 0.09333 and 0.07121 respectively (training set).

(b) KDE’s for transformed FPKM signal and the random forest fit thereof, with bandwidths 0.10109 and 0.07724 respectively (testing set).

Figure C.1: Plots for transformed expression data resulting from a random forest.

C.2 Support Vector Regression

The “svm” function from the “e1071” by Meyer et al (2015) [59] was used for performing SVR. All tuning parameters were left at their default values, and the output provided below:

Call:
svm(formula = FPKM_lambda ~ H3K4me3_signal_bin25 +
     H3K27me3_signal_bin21,
     data = training)

Parameters:
Appendix C. Machine Learning

**SVM-Type:** eps-regression  
**SVM-Kernel:** radial  
  - **cost:** 1  
  - **gamma:** 0.5  
  - **epsilon:** 0.1  

**Number of Support Vectors:** 16836

Figure C.2 compares the KDE’s for the dependent variable, with the predictions resulting from SVR. In each case, the smoothing parameter is also reported.

(a) KDE’s for transformed FPKM signal and the support vector regression fit thereof, with bandwidths 0.09333 and 0.08369 respectively (training set).  
(b) KDE’s for transformed FPKM signal and the support vector regression fit thereof, with bandwidths 0.10109 and 0.09052 respectively (testing set).

**Figure C.2:** Plots for transformed expression data resulting from support vector regression.

### C.3 Summary

Comparison of Table C.1 with Table 4.2 clearly indicates that neither the RF, nor SVR yields superior results, without serious parameter tuning. Note that the number of proportions used in the Bhattacharyya test, were reduced to 5, and may therefore produce
Appendix C. Machine Learning

<table>
<thead>
<tr>
<th></th>
<th>RF</th>
<th>SVR</th>
</tr>
</thead>
<tbody>
<tr>
<td>r² (Train)</td>
<td>0.4690</td>
<td>0.4616</td>
</tr>
<tr>
<td>r² (Test)</td>
<td>0.4718</td>
<td>0.4593</td>
</tr>
<tr>
<td>D2 (Train)</td>
<td>0.2010</td>
<td>0.2507</td>
</tr>
<tr>
<td>D2 (Test)</td>
<td>0.2069</td>
<td>0.2518</td>
</tr>
</tbody>
</table>

**Table C.1:** Comparison of performance results for random forest and support vector regression methods.

an artificially small divergence measure; for larger values, this measure couldn’t be computed because of empty cells.
# Appendix D

## Acronyms

The acronyms used in this thesis are listed in this appendix. The list is arranged alphabetically and the respective acronyms typeset in bold, with its corresponding meaning alongside.

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>AD</td>
<td>Anderson-Darling</td>
</tr>
<tr>
<td>ALR</td>
<td>arc length regression</td>
</tr>
<tr>
<td>CDF</td>
<td>cumulative distribution function</td>
</tr>
<tr>
<td>CvM</td>
<td>Cramér-von Mises</td>
</tr>
<tr>
<td>GoF</td>
<td>goodness-of-fit</td>
</tr>
<tr>
<td>IID</td>
<td>independent and identically distributed</td>
</tr>
<tr>
<td>KDE</td>
<td>kernel density estimate</td>
</tr>
<tr>
<td>ML</td>
<td>maximum-likelihood</td>
</tr>
<tr>
<td>MLE</td>
<td>maximum-likelihood estimation</td>
</tr>
<tr>
<td>MMM</td>
<td>moment matching</td>
</tr>
<tr>
<td>NALR</td>
<td>non-linear arc length regression</td>
</tr>
<tr>
<td>NLS</td>
<td>non-linear least squares</td>
</tr>
<tr>
<td>OLS</td>
<td>ordinary least squares</td>
</tr>
<tr>
<td>Acronym</td>
<td>Definition</td>
</tr>
<tr>
<td>---------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>PDF</td>
<td>probability density function</td>
</tr>
<tr>
<td>QF</td>
<td>quantile function</td>
</tr>
<tr>
<td>RF</td>
<td>random forest</td>
</tr>
<tr>
<td>rMSE</td>
<td>root-mean-square error</td>
</tr>
<tr>
<td>RSS</td>
<td>residual standard error</td>
</tr>
<tr>
<td>SVR</td>
<td>support vector regression</td>
</tr>
<tr>
<td>TLS</td>
<td>total least squares</td>
</tr>
</tbody>
</table>
Appendix E

Symbols

In this appendix, the symbols used throughout the thesis are defined, under the chapter in which they first appear.

E.1 Chapter 2: Preliminaries

\[ S_f^{[a,b]} \quad \text{Theoretical arc length} \quad \text{[Eq. (2.1.1), pg. 7]} \]

E.2 Chapter 3: Estimation

\[ f^{[a,b]} \quad \text{Discrete sample arc length statistic} \quad \text{[Eq. (3.2.1), pg. 11]} \]

\[ 2f^{[a,b]} \quad \text{Continuous sample arc length statistic} \quad \text{[Eq. (3.2.2), pg. 12]} \]

E.3 Chapter 5: Non-Linear Arc Length Regression

\[ f(a,b) \quad \text{Sample arc length statistic} \quad \text{[Eq. (5.1.1), pg. 42]} \]
Appendix F

Derived Publications

The following is a list of the publications derived from, and that are associated with, this thesis.


- Steven G. Hussey, Mattheüs T. Loots, Karen van der Merwe, Eshchar Mizrachi and Alexander A. Myburg. H3K4me3 and H3K27me3 in developing secondary xylem. *Scientific Reports* (Accepted for publication).


Index

Symbols
χ² approximation .......................... 38

A
AD ................................. see Anderson-Darling
alE ..................................... 78, 80, 82
alEdist ................................. 81, 82
alEffit (alE) ............................ 78
alEffitdist (alE) ........................ 78
alEtest (alEdist) ......................... 81
alKDEboot .............................. 87
ALR .............................. see arc length regression
Anderson-Darling 11, 15, 16, 18, 19, 21, 24, 25,
  67, 68, 100, 113
arc length 2, 7, 31, 42, 60, 61, 63–65, 69
arc length regression 4, 5, 31, 35, 36, 39, 68, 69
arc length test 5, 10, 19, 21, 24, 25, 67, 68

B
bhatt.test ................................. 91
Bhattacharyya divergence test 37, 39, 68, 123
bias ............................... 4, 15, 16, 24, 67, 100, 112
bootstrap .......................... 31, 34, 35, 39, 54, 68
Burr type III distribution .... 52, 55, 59
bw ........................................ 79, 82, 85, 88
C
c++ ..................................... 8
CDF ................................. see cumulative distribution function
comm.timer ........................... 86, 87, 90, 94, 95, 98
constrained optimisation ............... 55, 57, 59
Cramér-von Mises 10, 15, 16, 18, 19, 21, 24, 25,
  67, 68, 100, 113
cumulative distribution function 41, 43, 44, 47,
  53, 59–61, 63, 68
cumulative hazard function ................. 61
CvM .................................. see Cramér-von Mises

D
Deming regression .................... 26
differential evolution .................... 54
directional statistics .................... 2
e
errors-in-variables model .............. 26
Euclidean distance ...................... 15, 30, 31
exact distribution ...................... 38
exponential distribution ............... 52, 60
exponential power distribution ....... 22, 22

F
first-order linear ordinary inhomogeneous
differential equation .................. 61
four-parameter kappa distribution 47, 50, 53,
  55, 56, 59, 68, 69
Fredholm equation ..................... 69

G
Gaussian kernel ....................... 11, 28, 30, 33, 34, 36
generalised extreme value distribution 52
generalised logistic distribution ....... 52
generalised Pareto distribution ....... 52
global optimisation .................... 54
GoF .................................. see goodness-of-fit
goodness-of-fit 2, 4, 10, 11, 18, 19, 21, 24, 25,
  60, 67, 68
Gumbel distribution ...................... 52

H
hazard function ........................ 60, 61, 63

© University of Pretoria
Index

heatmap ........................................... 11, 19, 20

I
IID see independent and identically distributed independent and identically distributed ... 2, 15

J
JDEoptim ........................................ 94, 97

K
kappa4alBoot .................................... 96
kappa4nlsBoot ................................... 92
KDE  see kernel density estimate
kernel density estimate .................................. 11, 12, 28, 33, 34, 36, 37, 39, 68, 69, 122, 123

L
Laplace distribution .................................. 22
Lengau ............................................. 8, 11
leptokurtic alternatives .......................... 22, 24
linear model ....................................... 39, 68

M
maximum goodness-of-fit estimator ............. 10
maximum likelihood .......................... 27
maximum-likelihood estimation .... 15, 16, 24, 67, 100, 113
mesokurtic deviations .......................... 19, 22
method of arc lengths .... 5, 10, 15, 16, 19, 24, 67, 68, 100, 112, 113
minimum distance estimator .................. 10
minimum variance ................................ 27, 38
ML  see maximum-likelihood estimation
MLE  see maximum-likelihood estimation
mmKDEboot ........................................ 84
MMM  see moment matching method
mode .............................................. 43, 48, 53, 55, 57
moment matching ............................... 29, 69
moment matching method .... 31, 33, 34, 39, 68
monotone failure rate .......................... 60

N
NALR  see non-linear arc length regression
Nelder-Mead ........................................ 31
NLS  see non-linear least squares
non-linear arc length regression ... 4, 41, 54–57, 59, 68, 69
non-linear constraints .......................... 54
non-linear least squares 5, 41, 42, 54–57, 59, 68
non-parametric regression ................ 28
normal distribution .................. 7, 15, 19, 22, 69
null distribution ............................. 25, 67
null hypothesis .................. 15, 18, 20, 21, 24
objective function 15, 15, 30, 31, 39, 41, 42, 54, 68
OLS  see ordinary least squares
optim ........................................... 79, 85, 89
order statistics ................................. 3, 11
ordinary least squares .... 5, 26, 27, 31–33, 38, 39, 41, 68
orthogonal regression ..................... 26

P
parameter constraints .................. 5, 68, 69
parameter estimation .... 4, 10, 11, 15, 19, 24, 26, 37, 60, 67, 68
parametric bootstrap .................. 10, 18, 20, 25, 67
PDF  see probability density function
percentage variation explained .... 32, 39, 68
platykurtic alternatives ......... 22, 24, 25, 67
power exponential distribution ........ 11
predict.alKDEboot (alKDEboot) ........ 87
predict.kappa4alBoot (kappa4alBoot) .... 96
predict.kappa4nlsBoot (kappa4nlsBoot) 92
predict.mmKDEboot (mmKDEboot) ........ 84
print.alEfit (alE) ............................. 78
print.alEtest (alEdist) ..................... 81
print.alKDEboot (alKDEboot) ........... 87
print.kappa4alBoot (kappa4alBoot) .... 96
print.kappa4nlsBoot (kappa4nlsBoot) 92
print.mmKDEboot (mmKDEboot) ........ 84
print.summary.alKDEboot (alKDEboot) ............... 87
print.summary.kappa4alBoot (kappa4alBoot) ........... 96
print.summary.kappa4nlsBoot (kappa4nlsBoot) .......... 92
print.summary.mmKDEboot (mmKDEboot) .................. 84
probability density function 7, 8, 10–12, 19, 22, 43, 47, 50, 60, 61, 63, 64, 68, 69
probability integral transform ................. 2, 69
Q
QF. see quantile function
quantile function .................. 47, 68
R
random forest .................. 121–123
residual standard error .................. 55, 57
reverse exponential distribution ............... 52
RF. see random forest
rMSE. see root-mean-square error
root-mean-square error 4, 15, 16, 24, 55, 57, 67, 100, 112, 113
RSS. see residual standard error
S
saddlepoint approximation .................. 2, 3
sample arc length statistics 2, 3, 10, 12, 14–16, 20, 23–25, 60, 67, 100, 112
sample quantiles .................. 15, 100
shifted exponential distribution ............... 64
sigmoidal function .................. 5, 52
skew alternatives ............. 18, 19, 21, 23, 25, 67
skew normal distribution ................. 11, 19
smoothing parameter ... 12, 28, 33, 34, 36, 122, 123
spacings .................. 2, 3, 10, 60, 69
standard normal distribution .................. 44, 45
summary.alKDEboot (alKDEboot) ............... 87
summary.kappa4alBoot (kappa4alBoot) .......... 96
summary.kappa4nlsBoot (kappa4nlsBoot) ...... 92
summary.mmKDEboot (mmKDEboot) ............... 84
supercomputer .................. 8, 11
support vector regression .............. 121–123
SVR. see support vector regression

T
task-pull parallelism .................. 8
test statistic .................. 18
TLS. see total least squares
total least squares .................. 26
truncated distribution .............. 53, 55, 57, 59, 68
tuning parameter .................. 5, 8, 42, 67, 122, 123
Type I error .................. 23

U
U-statistics .................. 2
unbiased .................. 27, 38
uniform distribution .................. 22, 52, 60, 64

V
violin plot .................. 11, 12

W
wrapped distributions .................. 2